THEORY AND APPLICATIONS OF THE SINE-GABOR WAVELET FRAME

By

SERGIO SCHULER

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

1997

In memory of Dr. Kermit Sigmon.

ACKNOWLEDGEMENTS

First and foremost, I wish to thank my father, Karl Schuler and my mother, Svetlana Schuler, for their endless support and encouragement.

I would like to thank my advisor Dr. Andrew Laine for his support during my years as a student in the master's and Ph.D. programs at the Computer and Information Science and Engineering department. Also, I would like to express my gratitude to Iztok Koren for the technical discussions we shared throughout our stay at the University of Florida and for sharing his efficient dyadic wavelet transform C code which serves as a base for the multivoice discrete wavelet transform developed in this dissertation.

My sincere thanks are expressed to John Bowers for his advice regarding the intricacies of graduate school as well as for sponsoring and nominating me for various academic awards.

Last, but not least, I am very thankful to my wife, Teresa Da Silva, my son, Carlos Eduardo, and my daughter Ana Cristina for their understanding and patience during the extent of my studies. I am extremely grateful for their love.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

THEORY AND APPLICATIONS OF THE SINE-GABOR WAVELET FRAME

By

Sergio Schuler

August 1997

Chairman: Dr. Andrew F. Laine
Major Department: Computer and Information Science and Engineering

Nonorthogonal wavelet transforms play an important role in signal processing by offering much greater flexibility in the choice of analyzing wavelet than their orthogonal counterparts; however, very little literature has been devoted to their discrete implementation. In this work we introduce a new nonorthogonal wavelet, the sine-Gabor function, and propose an invertible discrete implementation of the wavelet transform using this new analyzing wavelet.

We show that the sine-Gabor function not only satisfies the wavelet admissibility condition but achieves nearly optimum time-frequency localization and generates a frame of wavelets. Furthermore, we show that there is a trade-off between the tightness of the frame and the time-frequency localization properties of the sine-Gabor wavelet. We demonstrate how this trade-off can be overcome by introducing voices. We also show that under certain conditions the sine-Gabor wavelet is nearly equal to the first derivative of a Gaussian and generates a "snug" frame. This implies that the first derivative of a Gaussian (Canny's approximation to the optimal step edge

detector) can be used to completely characterize and reconstruct a signal from its discrete wavelet coefficients. We address significant technical issues regarding the discretization of the wavelet transform and its computation when using nonorthogonal wavelets. We show that there exists a class of wavelets (the sine-Gabor wavelet being one of them) for which the discrete wavelet transform is not invertible. We describe how to evade this problem by modifying the strategy for selecting the discrete filters implementing the transform. We show how in the case of the sine-Gabor wavelet, desirable properties such as constant phase and nearly optimum time-frequency localization are preserved by the alternative filters. We conclude by extending the transform to two dimensions and demonstrating its use in image processing with both broad-band and narrow-band sine-Gabor wavelets and as a multivoice multiresolution edge detector.

CHAPTER 1
INTRODUCTION

The analysis of nonstationary signals often involves a compromise between how well transitions can be located, and how finely long-term behavior can be identified [Vet92]. Wavelets are rapidly finding application as a tool for the analysis of such signals [Com90, Dau92]. However, research has been primarily devoted to orthogonal and biorthogonal wavelet transforms because of their efficient and exact discrete implementations [Dau88, Mal89a, Mal89b, Mal89c, Dau90, Vet92]. Nonorthogonal wavelet transforms play an important role in signal processing by offering much greater flexibility in the choice of analyzing wavelet than their orthogonal counterparts. In particular, they allow adjustment of the relative resolution in time and scale [She96]. However, with some notable exceptions [She92, She93, She96], very little literature has been devoted to their discrete implementation. In this dissertation we introduce a new nonorthogonal wavelet, the sine-Gabor function, and extend the results of [She92, She93, She96] to obtain an invertible discrete implementation of the wavelet transform when using this new analyzing wavelet. Chapter 2 reviews the theory of the continuous wavelet transform and wavelet frames, listing a number of technical results that are used later in our study. Chapter 3 reviews some of the connections between discrete filters and wavelets with an emphasis on the algorithms proposed in [She92, She93, She96] for the computation of the discrete wavelet transform and its inverse for nonorthogonal wavelets. In Chapter 4 we introduce the sine-Gabor wavelet. We show that this function not only satisfies the wavelet admissibility condition but achieves nearly optimum time-frequency localization and

generates a frame of wavelets. Furthermore, we show that there is a trade-off between the tightness of the frame and the time-frequency localization properties of the sine-Gabor wavelet. We demonstrate how this trade-off can be overcome by introducing voices. We also show that under certain conditions the sine-Gabor wavelet is nearly equal to the first derivative of a Gaussian and generates a "snug" frame. This implies that the first derivative of a Gaussian (Canny's approximation to the optimal step edge detector [Can86]) can be used to completely characterize and reconstruct a signal from its discrete wavelet coefficients. In Chapter 5 we address the implementation issues regarding the computation of the discrete wavelet transform. We show that there exists a class of wavelets (the sine-Gabor wavelet being one of them) for which the discrete wavelet transform reviewed in Chapter 3 is not invertible. We describe how to evade this problem by modifying the strategy for selecting the discrete filters implementing the transform. We show how in the case of the sine-Gabor wavelet, desirable properties such as constant phase and nearly optimum time-frequency localization are preserved by the alternative filters. In Chapter 6 we investigate the application of the sine-Gabor wavelet transform to image processing. We describe an extension of the transform to two dimensions and demonstrate the use of the transform with both broad-band and narrow-band sine-Gabor wavelets having nearly optimum time-frequency localization. We also revisit the use of the sine-Gabor wavelet as an approximation to the first derivative of a Gaussian and demonstrate its use as a multivoice multiresolution edge detector. Finally, in Chapter 7 we summarize our work and describe possible extensions and open problems that are worthy of investigation.

CHAPTER 2
CONTINUOUS WAVELET TRANSFORM

In this chapter we review the continuous wavelet transform as well as wavelet frames. To highlight and contrast the advantages of the wavelet transform we also overview the short-time Fourier transform.

2.1 The Short-Time Fourier Transform

The goal of signal analysis is to extract relevant information from a signal by transforming it into a representation where the properties of the signal are more evident. For the analysis of stationary signals, that is, signals whose properties do not evolve with time, one example of such representation is the Fourier transform

$$(\mathcal{F}s)(\omega) = \int_{-\infty}^{\infty} s(t)e^{-j\omega t}\,dt.$$

The Fourier transform can be viewed as the inner product of the signal $s(t)$ and the sinusoidal wave $e^{j\omega t}$. The analysis coefficient $\hat{s}(\omega) = (\mathcal{F}s)(\omega)$ measures the "strength" of the sinusoidal wave of frequency ω in the signal $s(t)$.

Fourier analysis works well for signals composed of a few stationary components. However, any abrupt change in time in a nonstationary signal $s(t)$ is spread out over the whole frequency axis in $\hat{s}(\omega)$. In order to adapt the Fourier transform to nonstationary signals, Gabor [Gab46] introduced a new transform by using a window function $w(t)$ in the Fourier integral

$$(\mathcal{G}_w s)(\tau, \omega) = \int_{-\infty}^{\infty} s(t)\overline{w(t-\tau)}e^{-j\omega t}\,dt.$$

A function $w(t)$ qualifies as a window function if it is possible to identify its center and standard deviation (or root mean square (RMS) duration) defined by

$$m(w) = \frac{1}{\|w\|_2^2} \int_{-\infty}^{\infty} t|w(t)|^2 \, dt$$

and

$$\sigma(w) = \frac{1}{\|w\|_2} \left\{ \int_{-\infty}^{\infty} (t - m(w))^2 |w(t)|^2 \, dt \right\}^{1/2}, \tag{2.1}$$

respectively [Chu92]. In the original Gabor transform, the window function was a Gaussian [Gab46], however, the transform is valid for any type of window function and is referred to as the short-time or windowed Fourier transform.

The short-time Fourier transform can be viewed as the inner product of the signal $s(t)$ with the family of functions

$$w_{\tau,\omega}(t) = w(t - \tau)e^{j\omega t}, \ (\tau, \omega) \in \mathbb{R}^2,$$

that is,

$$(\mathcal{G}_w s)(\tau, \omega) = \langle s(t), w_{\tau,\omega}(t) \rangle \, . \tag{2.2}$$

It is easy to show that $w_{\tau,\omega}(t)$ is a window function with center and standard deviation given by

$$m(w_{\tau,\omega}) = m(w) + \tau \tag{2.3}$$

and

$$\sigma(w_{\tau,\omega}) = \sigma(w), \tag{2.4}$$

respectively. From Equations 2.2, 2.3 and 2.4 we have that in the time domain the analysis coefficient $(\mathcal{G}_w s)(\tau, \omega)$ essentially depends on the values of $s(t)$ for $t \in [m(w) + \tau - \sigma(w), m(w) + \tau + \sigma(w)]$. This is called time localization. It follows from the above observation that two pulses in time can be discriminated only if they are

more than $2\sigma(w)$ apart. This is referred to as the time resolution of the short-time Fourier transform.

By applying Parseval's theorem to Equation 2.2 we obtain an alternative formula for the short-time Fourier transform

$$(\mathcal{G}_w s)(\tau, \omega) = \frac{1}{2\pi} \langle \hat{s}(\gamma), \hat{w}_{\tau,\omega}(\gamma) \rangle \,, \tag{2.5}$$

where

$$\hat{w}_{\tau,\omega}(\gamma) = \hat{w}(\gamma - \omega) e^{-j(\gamma-\omega)\tau}, \ (\tau, \omega) \in \mathbb{R}^2.$$

Suppose $\hat{w}(\gamma) = (\mathcal{F}w)(\gamma)$ is also a window function with center, $m(\hat{w})$, and standard deviation, $\sigma(\hat{w})$. Then it is easy to show that $\hat{w}_{\tau,\omega}(\gamma)$ is a window function with center and standard deviation given by

$$m(\hat{w}_{\tau,\omega}) = m(\hat{w}) + \omega \tag{2.6}$$

and

$$\sigma(\hat{w}_{\tau,\omega}) = \sigma(\hat{w}), \tag{2.7}$$

respectively. From Equations 2.5, 2.6 and 2.7 we have that in the frequency domain the analysis coefficient $(\mathcal{G}_w s)(\tau, \omega)$ essentially depends on the values of $\hat{s}(\gamma)$ for $\gamma \in [m(\hat{w}) + \omega - \sigma(\hat{w}), m(\hat{w}) + \omega + \sigma(\hat{w})]$. This is called frequency localization. It follows from the above observation that two pure sinusoids can be discriminated only if their frequencies are more than $2\sigma(\hat{w})$ apart. This is referred to as the frequency resolution of the short-time Fourier transform.

From Equations 2.2 through 2.7 we have that $(\mathcal{G}_w s)(\tau, \omega)$ yields a time-frequency representation of $s(t)$. The analysis coefficient $(\mathcal{G}_w s)(\tau, \omega)$ depends on the time-frequency window

$$[m(w) + \tau - \sigma(w), m(w) + \tau + \sigma(w)] \times [m(\hat{w}) + \omega - \sigma(\hat{w}), m(\hat{w}) + \omega + \sigma(\hat{w})] \,.$$

Increasingly accurate localization in time and frequency is not possible because the area of the time-frequency window is lower bounded by 2, that is

$$4\sigma(w)\sigma(\hat{w}) \geq 2. \tag{2.8}$$

This is referred to as the uncertainty principle or Heisenberg inequality. Equality is satisfied if and only if the window function is a Gaussian as in the Gabor transform.

The short-time Fourier transform is an isometry (to a proportionality coefficient) from $L^2(\mathbb{R})$ into $L^2(\mathbb{R}^2)$ [Mal89c], that is

$$\|s\|_2^2 = \frac{1}{2\pi\|w\|_2^2} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} |(\mathcal{G}_w s)(\tau,\omega)|^2 \, d\tau d\omega.$$

The function $s(t)$ is reconstructed from $(\mathcal{G}_w s)(\tau,\omega)$ with the formula

$$s(t) = \frac{1}{2\pi\|w\|_2^2} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (\mathcal{G}_w s)(\tau,\omega) w(t-\tau) e^{j\omega t} \, d\tau d\omega.$$

The short-time Fourier transform is a redundant representation. Instead of computing $(\mathcal{G}_w s)(\tau,\omega)$ for all values (τ,ω) in $\mathbb{R}^2$, it is possible to uniformly sample both τ and ω such that the representation is complete and stable [Mal89c]. Let τ_0 and ω_0 be the sampling intervals in the time and frequency domains respectively. Then, the discrete short-time Fourier transform is defined by [Mal89c]

$$(\mathsf{G}_w s)_{m,n} = (\mathcal{G}_w s)(m\tau_0, n\omega_0), \ (m,n) \in \mathbb{Z}^2.$$

To reconstruct any function $s(t) \in L^2(\mathbb{R})$ from the set of samples $(\mathsf{G}_w s)_{m,n}$, $(m,n) \in \mathbb{Z}^2$, the operator

$$L^2(\mathbb{R}) \xrightarrow{\mathsf{G}_w} \ell^2(\mathbb{Z}^2)$$

must be invertible on its range and have a bounded inverse [Mal89c]. In order to invert G_w, Daubechies [Dau90] has shown that τ_0 and ω_0 must verify

$$\omega_0 \tau_0 < 2\pi.$$

A drawback of the short-time Fourier transform is that once a window function has been chosen the time-frequency resolution of the transform is fixed over the entire time-frequency plane. It is therefore impossible with the short-time Fourier transform to analyze at the same time transient signal components with good time resolution and quasi-stationary signal components with good frequency resolution [Rio91].

2.2 The Wavelet Transform

The wavelet transform overcomes the resolution limitation of the short-time Fourier transform by letting the time and frequency resolution vary in the time-frequency plane. The wavelet transform is defined by [Chu92]

$$(\mathcal{W}_\psi s)(a,b) = |a|^{-\frac{1}{2}} \int_{-\infty}^{+\infty} s(t) \overline{\psi\left(\frac{t-b}{a}\right)}\, dt,\ s(t) \in L^2(\mathbb{R}),\ a,b \in \mathbb{R},\ (a \neq 0). \quad (2.9)$$

In order to reconstruct $s(t)$ from its wavelet transform, the Fourier transform of $\psi(t)$ must satisfy [Chu92]

$$C_\psi = \int_{-\infty}^{+\infty} \frac{|\hat{\psi}(\gamma)|^2}{|\gamma|}\, d\gamma < \infty. \quad (2.10)$$

A function $\psi(t)$ is said to be a basic wavelet if it is possible to reconstruct $s(t)$ from its wavelet transform.

For any $\psi(t)$ satisfying Equation 2.10 we have that the wavelet transform is an isometry (to a proportionality coefficient) from $L^2(\mathbb{R})$ into $L^2(\mathbb{R}^2)$ [Chu92], that is

$$\|s\|_2^2 = \frac{1}{C_\psi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} |(\mathcal{W}_\psi s)(a,b)|^2\, \frac{da}{a^2} db. \quad (2.11)$$

The function $s(t)$ is reconstructed from $(\mathcal{W}_\psi s)(a,b)$, $a,b \in \mathbb{R}$, $(a \neq 0)$, with the formula [Chu92]

$$s(t) = \frac{1}{C_\psi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (\mathcal{W}_\psi s)(a,b) \left\{ |a|^{-\frac{1}{2}} \psi\left(\frac{t-b}{a}\right) \right\} \frac{da}{a^2} db.$$

The wavelet transform can be viewed as the inner product of the signal $s(t)$ with the family of functions

$$\psi_{a;b}(t) = |a|^{-\frac{1}{2}}\psi\left(\frac{t-b}{a}\right),\ a, b \in \mathbb{R},\ (a \neq 0), \tag{2.12}$$

that is,

$$(\mathcal{W}_\psi s)(a,b) = \langle s(t), \psi_{a;b}(t)\rangle\,. \tag{2.13}$$

Suppose $\psi(t)$ is a window function with center $m(\psi)$ and standard deviation $\sigma(\psi)$. Then, it is easy to show that $\psi_{a;b}(t)$ is a window function with center and standard deviation given by

$$m(\psi_{a;b}) = am(\psi) + b \tag{2.14}$$

and

$$\sigma(\psi_{a;b}) = |a|\sigma(\psi), \tag{2.15}$$

respectively. From Equations 2.13, 2.14 and 2.15 we have that in the time domain the analysis coefficient $(\mathcal{W}_\psi s)(a,b)$ essentially depends on the values of $s(t)$ for $t \in [am(\psi) + b - |a|\sigma(\psi), am(\psi) + b + |a|\sigma(\psi)]$. It can be verified [Chu92] that if a basic wavelet, $\psi(t)$, is also a window function then $\psi(t)$ is necessarily in $L^1(\mathbb{R})$ so that its Fourier transform, $\hat{\psi}(\omega)$ is a continuous function. It follows from Equation 2.10 that $\hat{\psi}(\omega)$ must vanish at the origin, that is,

$$\int_{-\infty}^{\infty} \psi(t)\,dt = 0. \tag{2.16}$$

By applying Parseval's theorem to Equation 2.13 we obtain an alternative formula for the wavelet transform

$$(\mathcal{W}_\psi s)(a,b) = \frac{1}{2\pi}\left\langle \check{s}(\gamma), \hat{\psi}_{a;b}(\gamma)\right\rangle, \tag{2.17}$$

where

$$\hat{\psi}_{a;b}(\gamma) = |a|^{\frac{1}{2}}\hat{\psi}(a\gamma)e^{-j\gamma b},\ a, b \in \mathbb{R},\ (a \neq 0).$$

Suppose $\hat{\psi}(\gamma)$ is also a window function with center, $m(\hat{\psi})$, and standard deviation, $\sigma(\hat{\psi})$. Then it is easy to show that $\hat{\psi}_{a;b}(\gamma)$ is a window function with center and standard deviation given by

$$m(\hat{\psi}_{a;b}) = \frac{m(\hat{\psi})}{a} \tag{2.18}$$

and

$$\sigma(\hat{\psi}_{a;b}) = \frac{\sigma(\hat{\psi})}{|a|}, \tag{2.19}$$

respectively. From Equations 2.17, 2.18 and 2.19 we have that in the frequency domain the analysis coefficient $(\mathcal{W}_\psi s)(a, b)$ essentially depends on the values of $\hat{s}(\gamma)$ for $\gamma \in \left[\frac{m(\hat{\psi})}{a} - \frac{\sigma(\hat{\psi})}{|a|}, \frac{m(\hat{\psi})}{a} - \frac{\sigma(\hat{\psi})}{|a|}\right]$.

From Equations 2.13 through 2.19 we have that $(\mathcal{W}_\psi s)(a, b)$ yields a time-frequency representation of $s(t)$. The analysis coefficient $(\mathcal{W}_\psi s)(a, b)$ depends on the time-frequency window

$$[am(\psi) + b - |a|\sigma(\psi), am(\psi) + b + |a|\sigma(\psi)] \times \left[\frac{m(\hat{\psi})}{a} - \frac{\sigma(\hat{\psi})}{|a|}, \frac{m(\hat{\psi})}{a} - \frac{\sigma(\hat{\psi})}{|a|}\right].$$

In contrast to the short-time Fourier transform, both the time and frequency resolution in the wavelet transform vary in the time-frequency plane. The following properties derive from the time-frequency localization characteristics of the wavelet transform:

1. The ratio of the standard deviation, $\sigma(\hat{\psi})/|a|$, of the frequency window and its center frequency, $m(\hat{\psi})/a$, is given by $\pm\frac{\sigma(\hat{\psi})}{m(\hat{\psi})}$, which is independent of the location of the center frequency. This is called constant-$\mathcal{Q}$ frequency analysis.

2. The time-frequency window narrows in time and widens in frequency for large center frequency $m(\hat{\psi})/a$ (small $a > 0$), and widens in time and narrows in frequency for small center frequency $m(\hat{\psi})/a$ (large $a > 0$).

Recall that the wavelet transform in Equation 2.9 can be viewed as the inner product of $s(t)$ with the family of functions $\psi_{a;b}(t)$ defined in Equation 2.12. Notice that this family of functions is obtained from a single basic wavelet by translating it by b and dilating it by a. The basic wavelet is usually referred to as the mother wavelet. The constant $|a|^{-\frac{1}{2}}$ in Equation 2.12 is used for energy normalization purposes, that is $\|\psi_{a;b}(t)\|_2^2 = \|\psi(t)\|_2^2$. In general the mother wavelet $\psi(t)$ is normalized to have its energy equal to one, that is $\|\psi(t)\|_2^2 = 1$. We will assume this restriction from now on.

In practice the parameter a is restricted to $a > 0$, so that the center frequency of $\hat{\psi}_{a;b}(\gamma)$ in Equation 2.18 is restricted to positive frequencies. In this context the parameter a is usually referred to as the scale parameter. In order to reconstruct $s(t)$ from its wavelet transform restricted to $a > 0$, the Fourier transform of $\psi(t)$ must satisfy a stricter admissibility condition than Equation 2.10 given by [Chu92]

$$\int_0^{+\infty} \frac{|\hat{\psi}(\gamma)|^2}{\gamma}\, d\gamma = \int_0^{+\infty} \frac{|\hat{\psi}(-\gamma)|^2}{\gamma}\, d\gamma = \frac{1}{2} C_\psi < \infty. \tag{2.20}$$

For any $\psi(t)$ satisfying Equation 2.20 we have the following reconstruction formula [Chu92]

$$s(t) = \frac{2}{C_\psi} \int_{-\infty}^{+\infty} \int_0^{+\infty} (\mathcal{W}_\psi s)(a,b) \left\{ \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a} \right) \right\} \frac{da}{a^2} db. \tag{2.21}$$

The wavelet transform is a redundant representation. Instead of computing $(\mathcal{W}_\psi s)(a,b)$ for $b \in \mathbb{R}$ and $a \in \mathbb{R}^+$, it is possible to choose an exponential sampling of the scale parameter, $a = a_0^j$, $j \in \mathbb{Z}$, $a_0 \neq 0$, such that the representation is complete and stable. Of particular interest is the sequence of scales where the elementary dilation step $a_0 = 2$ because it leads to an octave by octave partitioning of the frequency domain.

In order to reconstruct $s(t)$ from its wavelet transform restricted to $b \in \mathbb{R}$ and $a = a_0^j$, $j \in \mathbb{Z}$, the Fourier transform of $\psi(t)$ must satisfy a stricter condition than Equation 2.20 given by [Chu92]

$$A \leq \sum_{j=-\infty}^{+\infty} \left|\hat{\psi}(a_0^j \gamma)\right|^2 \leq B, \tag{2.22}$$

for almost all $\gamma \in \mathbb{R}$ for some constants A and B with $0 < A \leq B < \infty$. Reconstruction is possible by using the following formula [Chu92]

$$s(t) = \sum_{j=-\infty}^{+\infty} \int_{-\infty}^{+\infty} \left\{ a_0^{-\frac{j}{2}} (\mathcal{W}_\psi s)(a_0^j, b) \right\} \left\{ a_0^{-j} \tilde{\psi} \left(\frac{t-b}{a_0^j} \right) \right\} db,$$

where the Fourier transform of $\tilde{\psi}(t)$ is given by

$$\hat{\tilde{\psi}}(\gamma) = \frac{\hat{\psi}(\gamma)}{\sum_{j=-\infty}^{+\infty} \left|\hat{\psi}(a_0^j \gamma)\right|^2}.$$

If $\psi(t)$ satisfies the so-called "stability condition" in Equation 2.22 with $a_0 = 2$, then $\psi(t)$ is called a dyadic wavelet [Chu92].

In addition to sampling the scale parameter $a = a_0^j$, $j \in \mathbb{Z}$, it is possible to choose a sampling of the translation parameter b and still obtain a complete and stable representation. Intuitively, the discretization of b has to be chosen such that $\psi_{a;b}(t)$ "covers" the whole time axis for each scale $a = a_0^j$, $j \in \mathbb{Z}$. From Equation 2.15 we have that the standard deviation of $\psi_{a;b}(t)$ is $|a|$ times the standard deviation of $\psi(t)$. Hence, in order for $\psi_{a;b}(t)$ to "cover" the whole time axis at scale $a = a_0^j$, the discretization of b has to be proportional to $|a|$. It follows that at scale $a = a_0^j$ the translation has to be $b = k b_0 a_0^j$, where $b_0 > 0$ is the elementary translation step. This leads to a discretized family of wavelets given by

$$\psi_{j,k}(t) = a_0^{-\frac{j}{2}} \psi \left(\frac{t - k b_0 a_0^j}{a_0^j} \right) = a_0^{-\frac{j}{2}} \psi \left(a_0^{-j} t - k b_0 \right). \tag{2.23}$$

In order to reconstruct $s(t)$ from its wavelet transform restricted to $b = k b_0 a_0^j$, $j, k \in \mathbb{Z}$ and $a = a_0^j$, $j \in \mathbb{Z}$, $\psi(t)$ must satisfy a stricter condition than Equation 2.22. The admissibility condition for this reconstruction is the existence of

$A > 0$, $B < \infty$ so that

$$A\|s\|_2^2 \leq \sum_{j,k\in\mathbb{Z}} |\langle s(t), \psi_{j,k}(t)\rangle|^2 \leq B\|s\|_2^2$$

for all $s(t) \in L^2(\mathbb{R})$ [Dau92]. In other words, the family of functions $\psi_{j,k}(t)$ defined in Equation 2.23 constitute a frame. A brief review of frames is presented in Section 2.2.1.

Given a frame of wavelets, reconstruction of $s(t)$ from the $\langle s(t), \psi_{j,k}(t)\rangle$ is possible by using the following formula

$$s(t) = \sum_{j=-\infty}^{+\infty} \sum_{k=-\infty}^{+\infty} \langle s(t), \psi_{j,k}(t)\rangle \, \tilde{\psi}^{j,k}(t), \tag{2.24}$$

where the family of functions $\tilde{\psi}^{j,k}(t)$, $j, k \in \mathbb{Z}$ is called the dual frame of $\psi_{j,k}(t)$, $j, k \in \mathbb{Z}$ (see Section 2.2.1).

If the dual frame of $\psi_{j,k}(t)$, $j, k \in \mathbb{Z}$ is of the form

$$\tilde{\psi}^{j,k}(t) = \tilde{\psi}_{j,k}(t),\ j, k \in \mathbb{Z},$$

where

$$\tilde{\psi}_{j,k}(t) = a_0^{-\frac{j}{2}} \tilde{\psi}\left(\frac{t - kb_0a_0^j}{a_0^j}\right) = a_0^{-\frac{j}{2}} \tilde{\psi}\left(a_0^{-j}t - kb_0\right),$$

then Equation 2.24 is referred to as a wavelet series and $\tilde{\psi}(t)$ is known as the dual wavelet of $\psi(t)$. Given a wavelet frame, the existence of a dual wavelet is important for computational purposes. Below, we overview the classification of wavelets according to orthogonality and highlight the existence of a dual wavelet in each case. A wavelet can be classified according to orthogonality as follows:

1. A wavelet $\psi(t)$ is called semiorthogonal if it satisfies $\langle \psi_{j,k}(t), \psi_{l,m}(t)\rangle = 0$, $j \neq l$, $j, k, l, m \in \mathbb{Z}$.

   Remarks: It can be shown [Chu92] that for every semiorthogonal wavelet, $\psi(t)$, there exist a dual wavelet, $\tilde{\psi}(t)$, such that the pair $(\psi(t), \tilde{\psi}(t))$ satisfies the

biorthogonality property

$$\left\langle \psi_{j,k}(t), \tilde{\psi}_{l,m}(t) \right\rangle = \delta_{j,l}\delta_{k,m}, j, k, l, m \in \mathbb{Z}. \tag{2.25}$$

2. A wavelet $\psi(t)$ is called nonorthogonal if it is not a semiorthogonal wavelet.

3. A semiorthogonal wavelet $\psi(t)$ is called orthogonal if it satisfies $\langle \psi_{j,k}(t), \psi_{l,m}(t) \rangle = \delta_{j,l}\delta_{k,m}$, $j, k, l, m \in \mathbb{Z}$.

   Remarks: It can be shown that an orthogonal wavelet is self dual.

4. A nonorthogonal wavelet $\psi(t)$ is called biorthogonal if there exist a dual wavelet, $\tilde{\psi}(t)$, such that the pair $(\psi(t), \tilde{\psi}(t))$ satisfies the biorthogonality property in Equation 2.25.

The computation of Equation 2.24 can be a burden for nonorthogonal wavelets for which a dual wavelet does not exist. In such a case, it is advantageous to work with frames which are almost tight ("snug frames"), i.e., frames which have $B/A - 1 \ll 1$, because the $\tilde{\psi}^{j,k}(t)$ can be approximated by $\psi_{j,k}(t)$ (see Section 2.2.1 and [Dau92] for more details).

2.2.1 Frames

In this section we present a brief review of frames. For a more detailed and rigorous account of frames see [Dau92].

A family of functions φ_j, $j \in \mathbb{J}$ in a Hilbert space $\mathcal{H}$ is called a frame if there exist $A > 0$, $B < \infty$ so that for all $f \in \mathcal{H}$,

$$A\|f\|^2 \leq \sum_{j \in \mathbb{J}} |\langle f, \varphi_j \rangle|^2 \leq B\|f\|^2, \tag{2.26}$$

where A and B are called the frame bounds. If the two frame bounds are equal, the frame is called a tight frame. In a tight frame we have, for all $f \in \mathcal{H}$,

$$\sum_{j \in \mathbb{J}} |\langle f, \varphi_j \rangle|^2 = A\|f\|^2,$$

which implies

$$A\langle f, g\rangle = \sum_{j\in\mathbb{J}} \langle f, \varphi_j\rangle \langle \varphi_j, g\rangle ,$$

or (at least in the weak sense [Dau92]),

$$f = A^{-1} \sum_{j\in\mathbb{J}} \langle f, \varphi_j\rangle \varphi_j. \tag{2.27}$$

Although Equation 2.27 is reminiscent of the expansion of f into an orthonormal basis, it is important to note that frames, even tight frames, are not (orthonormal) bases. The family of functions φ_j, $j \in \mathbb{J}$ are typically not linearly independent. If the frame is tight, and if $\|\varphi_j\| = 1$ for all $j \in \mathbb{J}$, then $A = B$ gives the "redundancy ratio." If this ratio equals to 1, then the tight frame is an orthonormal basis [Dau92].

Equation 2.27 gives a trivial way to recover f from the $\langle f, \varphi_j\rangle$, if the frame is tight. Consider now recovering f from frames that are not tight. Let us define the frame operator F from $\mathcal{H}$ to $\ell^2(\mathbb{J})$ as

$$(Ff)_j = \langle f, \varphi_j\rangle .$$

Since φ_j constitute a frame, it follows from Equation 2.26 that $\|Ff\|^2 \leq B\|f\|^2$, that is, F is bounded, which means it is possible to find its adjoint operator F^* [Dau92]. The adjoint operator F^* of F can be computed from the following relation

$$\begin{aligned} \langle F^*c, f\rangle &= \langle c, Ff\rangle = \sum_{j\in\mathbb{J}} c_j \overline{\langle f, \varphi_j\rangle} \\ &= \sum_{j\in\mathbb{J}} c_j \langle \varphi_j, f\rangle , \end{aligned}$$

so that

$$F^*c = \sum_{j\in\mathbb{J}} c_j \varphi_j,$$

at least in the weak sense [Dau92]. From this it follows that

$$\sum_{j\in\mathbb{J}} \langle f, \varphi_j\rangle \varphi_j = F^*Ff.$$

Thus, the definition of F implies

$$\sum_{j\in\mathbb{J}} |\langle f, \varphi_j\rangle|^2 = \|Ff\|^2 = \langle F^*Ff, f\rangle .$$

Hence, in terms of F, the frame condition in Equation 2.26 can be written as

$$A\, I \leq F^*F \leq B\, I, \tag{2.28}$$

where I is the identity operator [Dau92]. This implies, in particular, that F^*F is invertible (see Lemma 3.2.2 in [Dau92]), and that the operator $(F^*F)^{-1}$ satisfies,

$$B^{-1}\, I \leq (F^*F)^{-1} \leq A^{-1}\, I.$$

Applying the operator $(F^*F)^{-1}$ to the family of functions φ_j, $j \in \mathbb{J}$, leads to another family of functions denoted by $\tilde{\varphi}_j$, $j \in \mathbb{J}$ where

$$\tilde{\varphi}_j = (F^*F)^{-1}\varphi_j,\ j \in \mathbb{J}.$$

The family of functions $\tilde{\varphi}_j$, $j \in \mathbb{J}$ also constitutes a frame with frame bounds B^{-1} and A^{-1} [Dau92], that is

$$B^{-1}\|f\|^2 \leq \sum_{j\in\mathbb{J}} |\langle f, \tilde{\varphi}_j\rangle|^2 \leq A^{-1}\|f\|^2.$$

It can be verified (see Proposition 3.2.3 in [Dau92]) that the associated frame operator $\tilde{F}$ from $\mathcal{H}$ to $\ell^2(\mathbb{J})$, $(\tilde{F}f)_j = \langle f, \tilde{\varphi}_j\rangle$ satisfies $\tilde{F} = F(F^*F)^{-1}$, $\tilde{F}^*\tilde{F} = (F^*F)^{-1}$, $\tilde{F}^*F = I = F^*\tilde{F}$ and $\tilde{F}F^* = F\tilde{F}^*$ is the orthogonal projection operator, in $\ell^2(\mathbb{J})$, onto $\mathrm{Ran}(F) = \mathrm{Ran}(\tilde{F})$. The family of functions $\tilde{\varphi}_j$, $j \in \mathbb{J}$ is called the dual frame of φ_j, $j \in \mathbb{J}$. It is easy to verify that the dual frame of $\tilde{\varphi}_j$, $j \in \mathbb{J}$ is φ_j, $j \in \mathbb{J}$. From $\tilde{F}^*F = I = F^*\tilde{F}$ it follows that

$$\sum_{j\in\mathbb{J}} \langle f, \varphi_j\rangle\, \tilde{\varphi}_j = f = \sum_{j\in\mathbb{J}} \langle f, \tilde{\varphi}_j\rangle\, \varphi_j. \tag{2.29}$$

Hence, we have a way to recover f from the $\langle f, \varphi_j\rangle$, where the only thing we need to do is to compute the $\tilde{\varphi}_j = (F^*F)^{-1}\varphi_j$. Note that, in general, if the frame is

redundant, there exist other functions in $\mathcal{H}$ that could equally well play the role of the $\tilde{\varphi}_j$ and lead to a reconstruction formula. This follows from the fact that the φ_j are not linearly independent in the general case. Equation 2.29, however, yields the most "economical" representation of f in the following sense (see Proposition 3.2.4 in [Dau92]): Consider the first half of Equation 2.29, $f = \sum_{j\in\mathbb{J}} \langle f, \varphi_j \rangle \tilde{\varphi}_j$. If $f = \sum_{j\in\mathbb{J}} \langle f, \varphi_j \rangle u_j$ for some other family of functions u_j, $j \in \mathbb{J}$, then it can be shown [Dau92] that $\sum_{j\in\mathbb{J}} |\langle f, u_j \rangle|^2 \geq \sum_{j\in\mathbb{J}} |\langle f, \tilde{\varphi}_j \rangle|^2$. Similarly, consider the second half of Equation 2.29, $f = \sum_{j\in\mathbb{J}} \langle f, \tilde{\varphi}_j \rangle \varphi_j$. If $f = \sum_{j\in\mathbb{J}} c_j \varphi_j$ for some $\boldsymbol{c} \in \ell^2(\mathbb{J})$, and if not all c_j equal $\langle f, \tilde{\varphi}_j \rangle$, then it can be shown [Dau92] that $\sum_{j\in\mathbb{J}} |c_j|^2 \geq \sum_{j\in\mathbb{J}} |\langle f, \tilde{\varphi}_j \rangle|^2$.

Computing the $\tilde{\varphi}_j$ involves the inversion of F^*F. If B is close to A ($r = B/A - 1 \ll 1$), then from Equation 2.28 we have that F^*F is "close" to $\frac{A+B}{2} I$. This implies that $(F^*F)^{-1}$ is "close" to $\frac{2}{A+B} I$, and that $\tilde{\varphi}_j$ is "close" to $\frac{2}{A+B}\varphi_j$ [Dau92]. This motivates the following reconstruction formula for f [Dau92],

$$f = \frac{2}{A+B} \sum_{j\in\mathbb{J}} \langle f, \varphi_j \rangle \varphi_j + Rf, \tag{2.30}$$

where

$$R = I - \frac{2}{A+B} F^*F. \tag{2.31}$$

It follows from Equations 2.28 and 2.31 that $-\frac{B-A}{B+A} I \leq R \leq \frac{B-A}{B+A} I$. This implies that $\|R\| \leq \frac{B-A}{B+A} = \frac{r}{2+r} < 1$. If r is small, the rest term Rf in Equation 2.30 can be dropped, leading to a reconstruction formula for f which is accurate up to an L^2-error of $\frac{r}{2+r}\|f\|$ [Dau92]. Even if r is not small, it is possible to write an algorithm for the reconstruction of f with exponential convergence [Dau92]. From Equation 2.31 we have that

$$F^*F = \frac{A+B}{2}(I - R).$$

This implies that

$$(F^*F)^{-1} = \frac{2}{A+B}(I - R)^{-1}.$$

Since $\|R\| < 1$, the series $\sum_{k=0}^{\infty} R^k$ converges in norm, and its limit is $(I - R)^{-1}$ [Dau92]. It follows that

$$\tilde{\varphi}_j = (F^*F)^{-1}\varphi_j = \frac{2}{A+B}\sum_{k=0}^{\infty} R^k \varphi_j. \tag{2.32}$$

Note that the zeroth order term in the above equation leads exactly to Equation 2.30 with the rest term dropped. Better approximations can be obtained by truncating after N terms [Dau92],

$$\tilde{\varphi}_j^N = \frac{2}{A+B}\sum_{k=0}^{N} R^k \varphi_j,$$

with

$$\left\| f - \sum_{j\in\mathbb{J}} \langle f, \varphi_j\rangle \, \tilde{\varphi}_j^N \right\| \leq \left(\frac{r}{2+r}\right)^{N+1} \|f\|,$$

which becomes exponentially small as N increases, since $\frac{r}{2+r} < 1$. In particular, the $\tilde{\varphi}_j$ can be computed by an iterative algorithm [Dau92],

$$\tilde{\varphi}_j^N = \frac{2}{A+B}\varphi_j + R\tilde{\varphi}_j^{N-1}.$$

Similarly, f can be computed by an iterative algorithm [Dau92],

$$f = (F^*F)^{-1}(F^*F)f = \lim_{N\to\infty} f_N,$$

with

$$f_N = f_{N-1} + \frac{2}{A+B}\sum_{j\in\mathbb{J}} \left[\langle f, \varphi_j\rangle - \langle f_{N-1}, \varphi_j\rangle\right]\varphi_j.$$

2.2.2 Wavelet Frames

Not all choices for $\psi(t)$, a_0, b_0 lead to frames of wavelets, even if $\psi(t)$ is admissible. Daubechies [Dau92] gives some general conditions on $\psi(t)$, a_0, b_0 under which a frame is obtained and derives estimates for the frame bounds. These results can be summarized as follows (see Proposition 3.3.2 in [Dau92]): If $\psi(t)$, a_0, are such that

$$\begin{aligned} \inf_{1\leq|\gamma|\leq a_0} \sum_{j=-\infty}^{+\infty} \left|\hat{\psi}(a_0^j\gamma)\right|^2 &> 0, \\ \sup_{1\leq|\gamma|\leq a_0} \sum_{j=-\infty}^{+\infty} \left|\hat{\psi}(a_0^j\gamma)\right|^2 &< \infty, \end{aligned} \tag{2.33}$$

and if

$$\beta(\gamma_s) = \sup_{1\leq|\gamma|\leq a_0} \sum_{j=-\infty}^{\infty} \left|\hat{\psi}(a_0^j\gamma)\right| \left|\hat{\psi}(a_0^j\gamma + \gamma_s)\right|$$

decays as fast as $(1+|\gamma_s|)^{-(1+\epsilon)}$, with $\epsilon > 0$, then there exist $b_{\text{thr}} > 0$ such that the $\psi_{j,k}(t)$ constitute a frame for all choices $b_0 < b_{\text{thr}}$. For $b_0 < b_{\text{thr}}$, the following expressions are frame bounds for the $\psi_{j,k}(t)$

$$A = \frac{1}{b_0}\left\{ \inf_{1\leq|\gamma|\leq a_0} \sum_{j=-\infty}^{+\infty} \left|\hat{\psi}(a_0^j\gamma)\right|^2 - \sum_{\substack{k=-\infty \\ k\neq 0}}^{+\infty} \left[\beta\left(\frac{2\pi}{b_0}k\right)\beta\left(-\frac{2\pi}{b_0}k\right)\right]^{1/2} \right\}.$$

$$B = \frac{1}{b_0}\left\{ \sup_{1\leq|\gamma|\leq a_0} \sum_{j=-\infty}^{+\infty} \left|\hat{\psi}(a_0^j\gamma)\right|^2 + \sum_{\substack{k=-\infty \\ k\neq 0}}^{+\infty} \left[\beta\left(\frac{2\pi}{b_0}k\right)\beta\left(-\frac{2\pi}{b_0}k\right)\right]^{1/2} \right\}.$$

The condition on $\beta(\gamma_s)$ and Equation 2.33 are satisfied if $|\hat{\psi}(\gamma)| \leq C|\gamma|^{\alpha}(1+|\gamma|)^{-\beta}$ with $\alpha > 0$, $\beta > \alpha + 1$. It can also be verified that if the $\psi_{j,k}(t)$ in Equation 2.23 constitute a frame with frame bounds A and B for some choice of a_0, b_0, then the Fourier transform of $\psi(t)$ satisfies

$$b_0 A \leq \sum_{j=-\infty}^{+\infty} \left|\hat{\psi}(a_0^j\gamma)\right|^2 \leq b_0 B \tag{2.34}$$

almost everywhere [Chu92].

As mentioned before it is advantageous to set the elementary dilation step $a_0 = 2$ because it leads to an octave by octave partitioning of the frequency domain. Hereafter, we will assume this partitioning of the frequency domain unless stated otherwise.

2.2.3 Multivoice Wavelet Frames

In Section 2.2 we mentioned that it is advantageous to use wavelet frames with $B/A - 1 \ll 1$, because the $\tilde{\psi}^{j,k}(t)$ can be approximated by $\psi_{j,k}(t)$. This implies that the sum in Equation 2.34 should be almost constant for $\gamma \neq 0$. This imposes

a strong restriction on $\psi(t)$ that is not generally satisfied. In order to overcome this problem without having to give up too much freedom in choosing $\psi(t)$ or its bandwidth, one can adopt the use of different voices per octave (i.e., suboctave sampling) [Dau92]. This can be achieved by using different wavelets $\psi_1(t), \psi_2(t), \ldots, \psi_N(t)$ and looking at the frame $\psi_{vjk}(t) = \frac{1}{\sqrt{2^j}}\psi_v\left(\frac{t}{2^j} - kb_0\right)$, $j, k \in \mathbb{Z}, v = 1, \ldots, N$. In [Dau92], Daubechies gives the following expressions for the frame bounds of this multivoice frame

$$A = \frac{1}{b_0}\left\{\inf_{1\le|\gamma|\le 2}\sum_{v=1}^{N}\sum_{j=-\infty}^{+\infty}|\hat{\psi}_v(2^j\gamma)|^2 - \sum_{k\neq 0}\sum_{v=1}^{N}\left[\beta_v\left(\frac{2\pi}{b_0}k\right)\beta_v\left(-\frac{2\pi}{b_0}k\right)\right]^{1/2}\right\},$$

$$B = \frac{1}{b_0}\left\{\sup_{1\le|\gamma|\le 2}\sum_{v=1}^{N}\sum_{j=-\infty}^{+\infty}|\hat{\psi}_v(2^j\gamma)|^2 + \sum_{k\neq 0}\sum_{v=1}^{N}\left[\beta_v\left(\frac{2\pi}{b_0}k\right)\beta_v\left(-\frac{2\pi}{b_0}k\right)\right]^{1/2}\right\},$$

with

$$\beta_v(\gamma_s) = \sup_{1\le|\gamma|\le 2}\sum_{j=-\infty}^{+\infty}|\hat{\psi}_v(2^j\gamma)||\hat{\psi}_v(2^j\gamma+\gamma_s)|.$$

It is suggested in [Dau92] that by choosing the $\hat{\psi}_1(\gamma), \ldots, \hat{\psi}_N(\gamma)$ to have slightly staggered frequency localization centers, coupled with good decay at ∞, one can achieve $B/A - 1 \ll 1$. One choice favored by several authors [Gro90, Dau92, Lee96] is to take "fractionally" dilated versions of a single wavelet

$$\psi_v(t) = 2^{-(v-1)/N}\psi(2^{-(v-1)/N}t). \tag{2.35}$$

In this case

$$\sum_{v=1}^{N}\sum_{j=-\infty}^{+\infty}|\hat{\psi}_v(2^j\gamma)|^2 = \sum_{j=-\infty}^{+\infty}|\hat{\psi}(2^{j/N}\gamma)|^2 \tag{2.36}$$

which can be made almost constant by choosing N large enough [Dau92].

CHAPTER 3
DISCRETE FILTERS AND WAVELETS

In this chapter we review the computation of the wavelet transform and its inverse for nonorthogonal wavelets. Specifically, we review the computation of the wavelet coefficients in equation 2.24 by using discrete filters and the inverse transform in a nonorthogonal multiresolution framework. We will refer to equation 2.24 loosely as a wavelet "series" even in the case that the dual frame is not generated by the dilations and translations of a single function. We will also use the term nonorthogonal wavelet loosely to refer to a nonorthogonal wavelet for which a wavelet dual does not exist. We also restrict the scope of our review to the case where the elementary dilation step $a_0 = 2$ and the elementary translation step $b_0 = 1$.

3.1 The À Trous and Mallat's Algorithms

Recall from Section 2.2 that the wavelet transform of a signal $s(t)$ is given by

$$(\mathcal{W}_\psi s)(a,b) = |a|^{-\frac{1}{2}} \int_{-\infty}^{+\infty} s(t)\overline{\psi\left(\frac{t-b}{a}\right)}\,dt,\ s(t) \in L^2(\mathbb{R}),\ a,b \in \mathbb{R},\ (a \neq 0).$$

The wavelet coefficients $\langle s(t), \psi_{j,k}(t)\rangle$ in Equation 2.24 for $a_0 = 2$, $b_0 = 1$ can be obtained by sampling the wavelet transform of $s(t)$ with $a = 2^j$, $j \in \mathbb{Z}$ and $b = 2^j k$, $j,k \in \mathbb{Z}$. We will refer to these coefficients as the decimated wavelet "series" coefficients and will denote them by

$$\begin{aligned}(\mathsf{C}_\psi s)_{j,k} &= (\mathcal{W}_\psi s)(2^j, 2^j k) \\ &= \langle s(t), \psi_{j,k}(t)\rangle\,, \end{aligned} \tag{3.1}$$

where $\psi_{j,k}(t) = \frac{1}{\sqrt{2^j}}\psi\left(\frac{t}{2^j} - k\right)$ is the generator of a family of decimated wavelets. We will refer to the operation of computing the above wavelet coefficients as the decimated wavelet transform.

For many applications the above sampling is inadequate. In particular, a finer grid is needed when computing the wavelet transform for applications that require "good" time-frequency localization. This can be achieved by computing an undecimated wavelet transform and the addition of voices [She92]. The undecimated wavelet transform of a signal $s(t)$ is obtained by sampling the wavelet transform of $s(t)$ with $a = 2^j$, $j \in \mathbb{Z}$ and $b = k$, $k \in \mathbb{Z}$. We will refer to these coefficients as the undecimated wavelet "series" coefficients and will denote them by

$$\begin{aligned} (\mathsf{R}_\psi s)_{j,k} &= (\mathcal{W}_\psi s)(2^j, k) \\ &= \left\langle s(t), \psi_k^j(t) \right\rangle, \end{aligned} \tag{3.2}$$

where $\psi_k^j(t) = \frac{1}{\sqrt{2^j}} \psi\left(\frac{t-k}{2^j}\right)$ is the generator of a family of undecimated wavelets.

In what follows we will review two separately motivated implementations of the wavelet transform, the *algorithme à trous* [Hol90] and Mallat's [Mal89a] multiresolution decomposition. These algorithms are both special cases of a single filter bank structure, the discrete wavelet transform [She92], the behavior of which is governed by one's choice of filters. The à trous algorithm was originally devised as a computationally efficient implementation of the wavelet transform; however, it is more properly viewed as a nonorthogonal multiresolution decomposition for which the discrete wavelet transform computes a "sampled" continuous wavelet transform (wavelet "series" coefficients) exactly [She92]. On the other hand, Mallat's algorithm was devised as an orthogonal multiresolution decomposition [Mal89a] that when initialized properly computes a "sampled" continuous wavelet transform [She92]. For a more detailed and rigorous account of these algorithms see [Mal89a, Hol90, She92].

The following additional terminology and notation will be used throughout this chapter. The discretized wavelet transform (with sampling period $T = 1$) is given by

$$(w_\psi s)(a, b) = \frac{1}{\sqrt{a}} \sum_n s(n)\, \overline{\psi\left(\frac{n-b}{a}\right)}. \tag{3.3}$$

The discretized decimated wavelet "series" coefficients are given by

$$(\mathsf{c}_\psi s)_{j,k} = (w_\psi s)(2^j, 2^j k) = \frac{1}{\sqrt{2^j}} \sum_n s(n)\, \overline{\psi\left(\frac{n}{2^j} - k\right)}. \tag{3.4}$$

Similarly, the discretized undecimated wavelet "series" coefficients are given by

$$(\mathsf{r}_\psi s)_{j,k} = (w_\psi s)(2^j, k) = \frac{1}{\sqrt{2^j}} \sum_n s(n)\, \overline{\psi\left(\frac{n-k}{2^j}\right)}. \tag{3.5}$$

Signals and filters in bold face will be treated as vectors. The kth element of vector $\boldsymbol{f}$ is denoted by $[\boldsymbol{f}]_k = f_k$. The symbol † will be used for the mirror filter $[\boldsymbol{f}^\dagger]_k = f_k^\dagger = \overline{f}_{-k}$. The decimator operator is denoted by a matrix

$$\begin{aligned} D_{k,m} &= \delta(2k - m) \\ &= \delta_{2k,m} \end{aligned}$$

where $\delta_{k,m}$ is the Kronecker delta and $[\boldsymbol{\delta}]_k = \delta(k) = \delta_{k,0}$. The dilation operator is denoted by a matrix

$$\begin{aligned} U_{k,m} &= \delta(k - 2m) \\ &= \delta_{k,2m}. \end{aligned}$$

Convolution of $\boldsymbol{f}$ and $\boldsymbol{s}$ is denoted by $[\boldsymbol{f} * \boldsymbol{s}]_k = \sum_m f_{k-m} s_m$. Convolution followed by decimation becomes $[\boldsymbol{D}(\boldsymbol{f} * \boldsymbol{s})]_k = \sum_m D_{k,m}[\boldsymbol{f} * \boldsymbol{s}]_m = [\boldsymbol{f} * \boldsymbol{s}]_{2k} = \sum_m f_{2k-m} s_m$. The following shorthand will be used for the convolution of $\boldsymbol{f}$ and $\boldsymbol{s}$. $\boldsymbol{f} * \boldsymbol{s} = \boldsymbol{F}\boldsymbol{s}$, where $F_{m,n} = f_{m-n}$. It is easy to verify that $\boldsymbol{D}(\boldsymbol{f} * \boldsymbol{s}) = \boldsymbol{D}\boldsymbol{F}\boldsymbol{s}$. The symbol † will also be used for the Hermitian transpose of matrices $[\boldsymbol{F}^\dagger]_{m,n} = \overline{f}_{n-m} = f_{m-n}^\dagger$. For discrete signals the translation operator is denoted as

$$[T_m \boldsymbol{s}]_n = s_{m-n},$$

and for continuous signals as

$$(T_\tau s)(t) = s(t - \tau).$$

The z transform of a discrete signal s_n is given by

$$\check{s}(z) = \sum_n s_n z^{-n}.$$

Finally, we state the following two identities [She92], which will be used later. For any $\boldsymbol{F}$ of the form $F_{m,n} = f_{m-n}$ we have that

$$[(\boldsymbol{DF})^j]_{n,k} = [(\boldsymbol{DF})^j]_{0,k-2^j n} \tag{3.6}$$

and

$$\sum_k [(\boldsymbol{DF})^j]_{n,k} z^k = z^{2^j n} \prod_{r=0}^{j-1} \check{f}(z^{2^r}). \tag{3.7}$$

3.1.1 The Decimated À Trous Algorithm

The à trous algorithm was originally devised as an efficient implementation for computing wavelet series coefficients. To achieve this, the wavelet series coefficients in Equations 3.1 and 3.2 are approximated by their discretized counterparts in Equations 3.4 and 3.5, respectively. As a starting point consider implementing Equation 3.4. Clearly, as j increases $\psi(t)$ must be sampled at progressively more points, creating a large computational burden [Hol90, She92]. The solution proposed by [Hol90] is to approximate nonintegral points via a finite filter $\boldsymbol{f}^\dagger$. As an example, let $\boldsymbol{f}^\dagger$ be the filter $\frac{1}{\sqrt{2}}(0.5, 1.0, 0.5)$. Then,

$$\begin{aligned} \sqrt{2} \sum_k f^\dagger_{n-2k} \psi(k) &= \begin{cases} \psi(\frac{n}{2}), & \text{for } n \text{ even}, \\ \frac{1}{2}\left(\psi(\frac{n-1}{2}) + \psi(\frac{n+1}{2})\right), & \text{for } n \text{ odd}, \end{cases} \\ &\approx \psi\left(\frac{n}{2}\right), \end{aligned}$$

approximates a sampling of $\psi(t/2)$. Let $\boldsymbol{g}$ be a filter defined by $g_n = \overline{\psi(-n)}$. Then the above interpolation can be computed by first dilating $\boldsymbol{g}^\dagger$ and then convolving the result with a filter $\boldsymbol{f}^\dagger$ which leaves the even points fixed and interpolates to get the odd points [She92]. The condition that $\boldsymbol{f}$ be the identity on even points is sufficiently important that merits the following definition: A lowpass filter $\boldsymbol{f}$ is said to be an à

trous filter if it satisfies

$$f_{2k} = \delta(k)/\sqrt{2}. \tag{3.8}$$

The $\sqrt{2}$ is simply a convenient means of including the normalization factor of Equation 3.4 in the filter.

It can be verified [She92] that the interpolation operation can be written as follows

$$\begin{aligned}
[\boldsymbol{f}^\dagger * (\boldsymbol{U}\boldsymbol{g}^\dagger)]_n &= [\boldsymbol{F}^\dagger \boldsymbol{U}\boldsymbol{g}^\dagger]_n \\
&= \sum_k f^\dagger_{n-2k} g^\dagger_k \\
&= \sum_k f^\dagger_{n-2k} \psi(k) \\
&\approx \frac{1}{\sqrt{2}} \psi\left(\frac{n}{2}\right).
\end{aligned}$$

This result and Equation 3.4 leads to the following approximation for the discretized decimated wavelet series coefficients at $j = 1$,

$$\begin{aligned}
(\mathsf{c}_\psi s)_{1,k} &= \frac{1}{\sqrt{2}} \sum_n s(n)\, \overline{\psi\left(\frac{n}{2} - k\right)} \\
&\approx \sum_n s_n \sum_m \overline{f}^\dagger_{n-2k-2m} \overline{g}^\dagger_m \\
&= \sum_m g_{k-m} \sum_n f_{2m-n} s_n \\
&= [\boldsymbol{g} * \boldsymbol{D}(\boldsymbol{f} * \boldsymbol{s})]_k \\
&= [\boldsymbol{G}\boldsymbol{D}\boldsymbol{F}\boldsymbol{s}]_k,
\end{aligned}$$

where $[\boldsymbol{s}]_n = s_n = s(n)$. Proceeding inductively, one can find the following approximation for the discretized decimated wavelet series coefficients for all j [She92]

$$(\mathsf{c}_\psi s)_{j,k} \approx [\boldsymbol{G}(\boldsymbol{D}\boldsymbol{F})^j \boldsymbol{s}]_k. \tag{3.9}$$

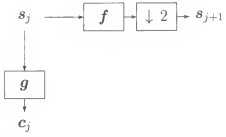


Figure 3.1: Block diagram of one stage of the discrete decimated wavelet transform.

Remarkably, the right-hand side in the above equation is the discrete decimated wavelet transform of $\boldsymbol{s}$ at level j [She92], which we denote as follows[1]

$$[\boldsymbol{c}_j(\boldsymbol{s})]_k = [\boldsymbol{G}(\boldsymbol{DF})^j \boldsymbol{s}]_k. \tag{3.10}$$

Equations 3.9 and 3.10 yield the following formula for the discretized decimated wavelet series coefficients

$$(c_\psi s)_{j,k} \approx [\boldsymbol{c}_j(\boldsymbol{s})]_k. \tag{3.11}$$

The discrete decimated wavelet transform can be computed iteratively over j as follows

$$\begin{aligned} [\boldsymbol{s}_{j+1}]_k &= \sum_n f_{2k-n}[\boldsymbol{s}_j]_n \\ &= [\boldsymbol{D}(\boldsymbol{f} * \boldsymbol{s}_j)]_k \end{aligned}$$

$$\begin{aligned} [\boldsymbol{c}_j]_k &= \sum_n g_{k-n}[\boldsymbol{s}_j]_n \\ &= [\boldsymbol{g} * \boldsymbol{s}_j]_k \end{aligned} \tag{3.12}$$

where $[\boldsymbol{s}_0]_k = s_k$. A block diagram of one stage of the transform is shown in Figure 3.1. Except for decimation of the output (see Section 3.1.3) the à trous algorithm described in [Hol90] is given by Equations 3.11 and 3.12.

The original à trous algorithm made no statements regarding the accuracy of the approximation in Equation 3.9 or even the discretization from Equation 2.9 to

[1]The notation $[\boldsymbol{c}_j]_k = c_{j,k}$ will be used as a shorthand for $[\boldsymbol{c}_j(\boldsymbol{s})]_k$.

Equation 3.3. A major step forward towards treating this question lies in the results of [She92] outlined in Section 3.2.

3.1.2 Mallat's Algorithm

In this section we review Mallat's multiresolution algorithm. For a more detailed and rigorous account of this algorithm see [Mal89a, Vet92, She92]. Mallat's algorithm has basically the same structure as Equations 3.12, that is,

$$\begin{aligned} [\boldsymbol{s}_{j+1}]_k &= [\boldsymbol{D}(\boldsymbol{h} * \boldsymbol{s}_j)]_k \\ [\boldsymbol{d}_{j+1}]_k &= [\boldsymbol{D}(\boldsymbol{g} * \boldsymbol{s}_j)]_k \end{aligned} \tag{3.13}$$

where $[\boldsymbol{s}_0]_k = s_k$. In keeping with the literature, the lowpass filter is denoted by $\boldsymbol{h}$ instead of $\boldsymbol{f}$. The constraints on $\boldsymbol{h}$ and $\boldsymbol{g}$ which ensure an orthonormal multiresolution analysis are [Mal89a, She92]:

1. Perfect reconstruction

$$\sum_j h_{2j-m}\overline{h}_{2j-n} + g_{2j-m}\overline{g}_{2j-n} = \delta_{m,n}, \tag{3.14}$$

2. Orthogonality of $\boldsymbol{h}$ and $\boldsymbol{g}$ with respect to even shifts

$$\sum_j h_{2m-j}\overline{g}_{2n-j} = 0, \tag{3.15}$$

3. Bandpass condition on $\boldsymbol{g}$

$$\sum_n g_n = 0, \tag{3.16}$$

4. Lowpass condition on $\boldsymbol{h}$

$$\sum_n h_n = \sqrt{2}. \tag{3.17}$$

In matrix notation Equations 3.14 and 3.15 may be written as follows

$$(\boldsymbol{H}^{\dagger}\boldsymbol{U})(\boldsymbol{D}\boldsymbol{H}) + (\boldsymbol{G}^{\dagger}\boldsymbol{U})(\boldsymbol{D}\boldsymbol{G}) = \boldsymbol{I}, \tag{3.18}$$

$$(\boldsymbol{D}\boldsymbol{H})(\boldsymbol{G}^{\dagger}\boldsymbol{U}) = \boldsymbol{0}, \tag{3.19}$$

where $H_{m,n} = h_{m-n}$. Multiplying Equation 3.18 on the left by $\boldsymbol{DH}$ and using Equation 3.19 we have that

$$(\boldsymbol{D}\boldsymbol{H})(\boldsymbol{H}^{\dagger}\boldsymbol{U}) = \boldsymbol{I}.$$

It follows that $\boldsymbol{h}$ and its shifted versions by even shifts form an orthonormal set

$$\sum_j h_{2m-j}\overline{h}_{2n-j} = \delta_{m,n},\ m, n \in \mathbb{Z}.$$

Similarly, multiplying Equation 3.18 on the left by $\boldsymbol{DG}$ and using Equation 3.19 we have that

$$(\boldsymbol{D}\boldsymbol{G})(\boldsymbol{G}^{\dagger}\boldsymbol{U}) = \boldsymbol{I}.$$

It follows that $\boldsymbol{g}$ and its shifted versions by even shifts also form an orthonormal set

$$\sum_j g_{2m-j}\overline{g}_{2n-j} = \delta_{m,n},\ m, n \in \mathbb{Z}.$$

These two results together with Equation 3.15 imply that Equations 3.13 is an orthogonal decomposition of the discrete signal $\boldsymbol{s}_j$. Furthermore, Equation 3.14 implies that

$$[\boldsymbol{s}_j]_k = [(\boldsymbol{H}^{\dagger}\boldsymbol{U})\boldsymbol{s}_{j+1} + (\boldsymbol{G}^{\dagger}\boldsymbol{U})\boldsymbol{d}_{j+1}]_k.$$

From Equations 3.14–3.17 it follows that Equations 3.13 represents a wavelet decomposition as described below [Mal89a, She92].

Define a scaling function $\phi(t)$ with Fourier transform given by

$$\hat{\phi}(\omega) = \prod_{r=1}^{\infty} \frac{1}{\sqrt{2}}\overline{\tilde{h}\left(e^{j\frac{\omega}{2^r}}\right)}.$$

It follows that

$$\hat{\phi}(\omega) = \frac{1}{\sqrt{2}} \overline{\hat{h}\left(e^{j\frac{\omega}{2}}\right)} \hat{\phi}\left(\frac{\omega}{2}\right)$$

which in the time domain takes the form

$$\phi(t) = \sum_k \overline{h}_{-k} \sqrt{2} \phi(2t - k).$$

Therefore, the dilates and translates of $\phi(t)$,

$$\phi_{j,k}(t) = \frac{1}{\sqrt{2^i}} \phi\left(\frac{t}{2^i} - n\right)$$

have the property [She92]

$$\phi_{j+1,k}(t) = \sum_n \overline{h}_{2k-n} \phi_{j,n}(t).$$

Define a wavelet function by

$$\psi(t) = \sum_k \overline{g}_{-k} \sqrt{2} \phi(2t - k). \tag{3.20}$$

Then, using Equation 3.15 and the above properties, one can show that the family of wavelets

$$\psi_{j,k}(t) = \frac{1}{\sqrt{2^i}} \psi\left(\frac{t}{2^i} - n\right)$$

are orthonormal, and that the $\boldsymbol{d}_j$ are the coefficients of the expansion of $s(t)$ in terms of $\psi_{j,k}(t)$, that is

$$[\boldsymbol{d}_j]_k = \frac{1}{\sqrt{2^j}} \int_{-\infty}^{+\infty} s(t) \overline{\psi\left(\frac{t}{2^j} - k\right)} dt,$$

provided that

$$[\boldsymbol{s}_0]_k = \int_{-\infty}^{+\infty} s(t) \overline{\phi(t - k)} dt.$$

Therefore, Mallat's algorithm when initialized properly (see the equation above) computes exactly a "sampled" continuous wavelet transform [She92].

3.1.3 The Undecimated À Trous Algorithm

As mentioned before for many applications the time-frequency sampling grid provided by the decimated wavelet transform is inadequate when "good" time-frequency localization is required. One way to overcome this limitation is by computing an undecimated wavelet transform. Following the same approach described above the wavelet series coefficients in Equation 3.2 can be approximated by their discretized counterpart in Equation 3.5. From Equations 3.4 and 3.5 it follows that $(\mathsf{c}_\psi s)_{j,k} = (w_\psi s)(2^j, 2^j k)$ and $(\mathsf{r}_\psi s)_{j,k} = (w_\psi s)(2^j, k)$ should coincide at $k = 0$. Utilizing this fact, one can obtain the kth undecimated wavelet series coefficient $(\mathsf{r}_\psi s)_{j,k}$ by translating the signal back by k and computing the discretized decimated wavelet series coefficient $\left(\mathsf{c}_\psi(T_{-k}s)\right)_{j,0}$. This result together with Equation 3.11 yields the following approximation for computing the discretized undecimated wavelet series coefficients

$$\begin{aligned}(\mathsf{r}_\psi s)_{j,k} &= \left(\mathsf{c}_\psi(T_{-k}s)\right)_{j,0} \\ &\approx [\boldsymbol{c}_j(T_{-k}\boldsymbol{s})]_0, \end{aligned} \tag{3.21}$$

where $[\boldsymbol{s}]_n = s_n = s(n)$. Remarkably, the right-hand side in the above equation is the discrete undecimated wavelet transform of $\boldsymbol{s}$ at level j [She92], which we denote as follows[2]

$$[\boldsymbol{r}_j(\boldsymbol{s})]_k = [\boldsymbol{c}_j(T_{-k}\boldsymbol{s})]_0. \tag{3.22}$$

Equations 3.21 and 3.22 yield the following formula for the discretized undecimated wavelet series coefficients

$$(\mathsf{r}_\psi s)_{j,k} \approx [\boldsymbol{r}_j(\boldsymbol{s})]_k. \tag{3.23}$$

[2]The notation $[\boldsymbol{r}_j]_k = r_{j,k}$ will be used as a shorthand for $[\boldsymbol{r}_j(\boldsymbol{s})]_k$

A few observations about the decimated and undecimated discrete wavelet transforms are in order. First notice that in general $\boldsymbol{c}_j$ is not translation invariant

$$\begin{aligned}[\boldsymbol{c}_j(T_m\boldsymbol{s})]_k &= \sum_n [\boldsymbol{G}(\boldsymbol{D}\boldsymbol{F})^j]_{k,n}[\boldsymbol{s}]_{n-m} \\ &= \sum_n [\boldsymbol{G}(\boldsymbol{D}\boldsymbol{F})^j]_{k,n+m}[\boldsymbol{s}]_n \\ &\neq \sum_n [\boldsymbol{G}(\boldsymbol{D}\boldsymbol{F})^j]_{k-m,n}[\boldsymbol{s}]_n.\end{aligned}$$

However, if one replaces m by $2^j m$ in the above equation and uses Equation 3.6, the last step becomes an equality [She92]

$$[\boldsymbol{c}_j(T_{2^j m}\boldsymbol{s})]_k = [\boldsymbol{c}_j(\boldsymbol{s})]_{k-m}. \tag{3.24}$$

Thus, translating $\boldsymbol{s}$ by $2^j m$, translates octave j by m.

On the other hand, it can be verified that $\boldsymbol{r}_j$ is translation invariant

$$\begin{aligned}[\boldsymbol{r}_j(T_m\boldsymbol{s})]_k &= [\boldsymbol{c}_j(T_{m-k}\boldsymbol{s})]_0 \\ &= [\boldsymbol{r}_j(\boldsymbol{s})]_{k-m}.\end{aligned}$$

Also, from Equations 3.22 and 3.24 it follows that sampling $r_{j,k}$ every 2^j points produces exactly $c_{j,k}$, that is,

$$c_{j,k} = r_{j,2^j k}.$$

An iterative algorithm for computing the discrete undecimated wavelet transform can be obtained by taking z transforms [She92]. From Equations 3.10 and 3.22

$$\begin{aligned}\sum_k r_{j,k} z^{-k} &= \sum_m \sum_k [\boldsymbol{G}(\boldsymbol{D}\boldsymbol{F})^j]_{0,m} s_{m+k} z^{-k} \\ &= \sum_m [\boldsymbol{G}(\boldsymbol{D}\boldsymbol{F})^j]_{0,m} z^m \check{s}(z).\end{aligned}$$

Applying Equation 3.7 to the above equation it follows that

$$\sum_k r_{j,k} z^{-k} = \check{g}(z^{2^j}) \prod_{r=0}^{j-1} \check{f}(z^{2^r}) \check{s}(z)$$

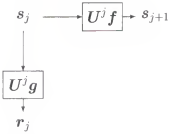


Figure 3.2: Block diagram of one stage of the discrete undecimated wavelet transform.

where $j = 0$ is understood to mean there are no factors of $\check{f}(z)$.

It is easy to see that $\boldsymbol{U}^j \boldsymbol{f}$ is $\boldsymbol{f}$ with $2^j - 1$ zeros inserted between every pair of filter coefficients and that its z transform is $\check{f}(z^{2^j})$. Therefore the discrete undecimated wavelet transform can be computed iteratively over j as follows

$$[\boldsymbol{s}_{j+1}]_k = [(\boldsymbol{U}^j \boldsymbol{f}) * \boldsymbol{s}_j)]_k$$

$$[\boldsymbol{r}_j]_k = [(\boldsymbol{U}^j \boldsymbol{g}) * \boldsymbol{s}_j]_k$$

where $[\boldsymbol{s}_0]_k = s_k$. The above result together with Equation 3.23 is essentially the à trous algorithm described in [Hol90]. A block diagram of one stage of the transform is shown in Figure 3.2.

3.2 The À Trous Algorithm as an Exact Wavelet Transform

In the previous section we analyzed two separately motivated implementations of the wavelet transform. We learned that the à trous algorithm was originally devised as a computationally efficient approximation of the "sampled" wavelet transform $(\mathsf{C}_\psi s)_{j,k}$ whose implementation can be viewed as a discrete wavelet transform for which the filter $\boldsymbol{f}$ is an interpolator satisfying Equation 3.8 and the filter $\boldsymbol{g}$ is obtained by sampling the wavelet function $\psi(t)$. We also learned that Mallat's algorithm is a discrete wavelet transform with filters $\boldsymbol{h}$ and $\boldsymbol{g}$ subject to the constraints in Equations 3.14–3.17 that when initialized properly computes exactly a "sampled" continuous wavelet transform for the wavelet function $\psi(t)$ in Equation 3.20. Next

we will review the results of [She92] which show that the à trous algorithm can be viewed as a nonorthogonal multiresolution decomposition for which the discrete wavelet transform computes a "sampled" continuous wavelet transform exactly.

Define a scaling function $\phi(t)$ with Fourier transform given by

$$\hat{\phi}(\omega) = \prod_{r=1}^{\infty} \frac{1}{\sqrt{2}} \overline{\hat{f}\left(e^{j\frac{\omega}{2^r}}\right)}, \tag{3.25}$$

or equivalently

$$\phi(t) = \lim_{j\to\infty} \sum_k [(\overline{\boldsymbol{DF}})^j]_{0,k} \sqrt{2^j} \chi(2^j t - k), \tag{3.26}$$

where

$$\chi(t) = \begin{cases} 1 & \text{for } t \in [-1/2, 1/2), \\ 0 & \text{otherwise.} \end{cases}$$

For normalization purposes $\hat{f}(e^{j\omega})\big|_{\omega=0} = \sqrt{2}$, which implies

$$\sum_k f_k = \sqrt{2} \tag{3.27}$$

and

$$\int_{-\infty}^{+\infty} \phi(t)dt = \hat{\phi}(0) = 1.$$

Suitable regularity conditions for the inverse Fourier transform of the product in Equation 3.25 to converge to a reasonably behaved function may be found in [Dau88, Mal89b, She92]. One can verify [She92] that the dilates and translates of $\phi(t)$ have the property

$$\begin{aligned} \phi_{j+1,k}(t) &= \sum_n [\overline{\boldsymbol{DF}}]_{k,n} \phi_{j,n}(t). \\ &= \sum_n \overline{f}_{2k-n} \phi_{j,n}(t). \end{aligned} \tag{3.28}$$

Furthermore, if $\boldsymbol{f}$ is a finite filter, then $\phi(t)$ has finite support [She92].

Define the wavelet function

$$\psi^{\dagger}(t) = \sum_k \overline{g}_k \phi(t+k) = \sum_k g_k^{\dagger} \phi(t-k). \tag{3.29}$$

We will refer to $\psi'(t)$ as the approximation of $\psi(t)$. The recursion in Equation 3.28 implies that

$$\phi_{j,k}(t) = \sum_n [(\overline{\boldsymbol{D}}\overline{\boldsymbol{F}})^j]_{k,n}\phi_{0,n}(t).$$

This expression along with Equation 3.29 yields

$$\psi'_{j,k}(t) = \sum_n [\overline{\boldsymbol{G}}(\overline{\boldsymbol{D}}\overline{\boldsymbol{F}})^j]_{k,n}\phi_{0,n}(t).$$

Using the above expression in Equation 3.4 we have that the discretized decimated wavelet "series" coefficients take the form

$$\begin{aligned}
(\mathsf{c}_{\psi'}s)_{j,k} &= \sum_l s(l)\overline{\psi'_{j,k}(l)} \\
&= \sum_l s(l) \sum_n [\overline{\boldsymbol{G}}(\overline{\boldsymbol{D}}\overline{\boldsymbol{F}})^j]_{k,n}\phi_{0,n}(l) \\
&= \sum_n [\boldsymbol{G}(\boldsymbol{D}\boldsymbol{F})^j]_{k,n} \sum_l s(l)\overline{\phi_{0,n}(l)} \\
&= \sum_n [\boldsymbol{G}(\boldsymbol{D}\boldsymbol{F})^j]_{k,n} \sum_l s(l)\overline{\phi(l-n)}.
\end{aligned}$$

Define

$$[\boldsymbol{s}]_n = \sum_l s(l)\overline{\phi(l-n)}, \tag{3.30}$$

then the above equation becomes

$$(\mathsf{c}_{\psi'}s)_{j,k} = [\boldsymbol{G}(\boldsymbol{D}\boldsymbol{F})^j\boldsymbol{s}]_k,$$

which is the discrete decimated wavelet transform of the discrete signal s_n defined in Equation 3.30. Therefore, using the initialization in Equation 3.30 the discrete decimated wavelet transform computes exactly the discretized decimated wavelet "series" coefficients $(\mathsf{c}_{\psi'}s)_{j,k}$. Similarly, using the above expression for $\psi'_{j,k}(t)$ in

Equation 3.1 we have that the decimated wavelet "series" coefficients take the form

$$\begin{aligned}
(\mathsf{C}_{\psi'} s)_{j,k} &= \int_{-\infty}^{+\infty} s(t)\overline{\psi'_{j,k}(t)}dt \\
&= \int_{-\infty}^{+\infty} s(t)\left(\sum_n [\boldsymbol{G}(\boldsymbol{DF})^j]_{k,n}\overline{\phi_{0,n}(t)}\right)dt \\
&= \sum_n [\boldsymbol{G}(\boldsymbol{DF})^j]_{k,n}\int_{-\infty}^{+\infty} s(t)\overline{\phi_{0,n}(t)}dt \\
&= \sum_n [\boldsymbol{G}(\boldsymbol{DF})^j]_{k,n}\int_{-\infty}^{+\infty} s(t)\overline{\phi(t-n)}dt.
\end{aligned}$$

Using

$$[\boldsymbol{s}]_n = \int_{-\infty}^{+\infty} s(t)\overline{\phi(t-n)}dt \tag{3.31}$$

the above equation becomes

$$(\mathsf{C}_{\psi'} s)_{j,k} = [\boldsymbol{G}(\boldsymbol{DF})^j \boldsymbol{s}]_k,$$

which is the discrete decimated wavelet transform of the discrete signal s_n defined in Equation 3.31. Therefore, using the initialization in Equation 3.31 the discrete decimated wavelet transform computes exactly the decimated wavelet "series" coefficients $(\mathsf{C}_{\psi'} s)_{j,k}$. Notice that Equation 3.30 is the discretized version of Equation 3.31.

Finally, we review the significance of the à trous condition in Equation 3.8. One may show that a filter $\boldsymbol{f}$ is an á trous filter *iff* $\phi(n) = \delta(n)$ [She92]. Using this observation it can be shown that if $\boldsymbol{f}$ is à trous then (1) Equation 3.30 becomes $[\boldsymbol{s}]_n = s(n)$ and the discrete decimated wavelet transform of the sampled signal $s(n)$ yields the exact discretized decimated wavelet "series" coefficients $(\mathsf{c}_{\psi'} s)_{j,k}$ and (2) the wavelet $\psi'(t)$ defined in Equation 3.29 satisfies $\psi'(n) = g_n^\dagger$ [She92].

In summary, given $\boldsymbol{f}$ satisfying Equations 3.8 and 3.27, $\boldsymbol{g}$ defined by $g_n^\dagger = \psi(n)$ (where $\psi(t)$ is an arbitrary wavelet), and given the initialization in Equation 3.31, the discrete wavelet transform computes an exact "sampled" continuous wavelet transform of $s(t)$ using the wavelet function $\psi'(t)$ defined in Equation 3.29. Furthermore,

if there is sufficient regularity, $\psi'(t)$ and $\psi(t)$ will be "close" since they coincide on the integers up to the length of $\boldsymbol{g}$.

3.3 The Undecimated À Trous Algorithm and Voices

As mentioned before we are interested in computing a multivoice undecimated wavelet transform. In this section we show how to extend the algorithm presented in Section 3.1.3 into the multivoice framework. Recall from Section 2.2.3 that in the multivoice case instead of having a basic wavelet $\psi(t)$ one deals with a family of wavelets $\psi_v(t)$, $v = 1, \ldots, N$. In the spirit of Equations 3.1 and 3.2 we denote the multivoice decimated and undecimated wavelet "series" coefficients as

$$(\mathsf{C}_{\psi_v} s)_{j,k} = (\mathcal{W}_{\psi_v} s)(2^j, 2^j k)$$

and

$$(\mathsf{R}_{\psi_v} s)_{j,k} = (\mathcal{W}_{\psi_v} s)(2^j, k),$$

respectively. Similarly, the discretized multivoice decimated and undecimated wavelet "series" coefficients are given by

$$(\mathsf{c}_{\psi_v} s)_{j,k} = (w_{\psi_v} s)(2^j, 2^j k) = \frac{1}{\sqrt{2^j}} \sum_n s(n)\, \overline{\psi_v\left(\frac{n}{2^j} - k\right)}$$

and

$$(\mathsf{r}_{\psi_v} s)_{j,k} = (w_{\psi_v} s)(2^j, k) = \frac{1}{\sqrt{2^j}} \sum_n s(n)\, \overline{\psi_v\left(\frac{n-k}{2^j}\right)},$$

respectively. Following the approach presented in Section 3.1.1 let $\boldsymbol{g}_v$ be the filter defined by $g_{v,n} = \overline{\psi_v(-n)}$ and let $\boldsymbol{f}$ be an à trous filter. Then

$$(\mathsf{c}_{\psi_v} s)_{j,k} \approx [\boldsymbol{c}_{vj}(\boldsymbol{s})]_k \tag{3.32}$$

where

$$[\boldsymbol{c}_{vj}(\boldsymbol{s})]_k = [\boldsymbol{G}_v (\boldsymbol{D}\boldsymbol{F})^j \boldsymbol{s}]_k. \tag{3.33}$$

Notice that Equation 3.33 resembles the structure of a discrete wavelet transform with the exception that we now have a family of filters $\boldsymbol{g}_v$. The coefficients $[\boldsymbol{c}_{v;j}]_k = [\boldsymbol{c}_{v;j}(\boldsymbol{s})]_k$ can be computed iteratively over j as follows

$$\begin{aligned}[\boldsymbol{s}_{j+1}]_k &= \sum_n f_{2k-n}[\boldsymbol{s}_j]_n \\ &= [\boldsymbol{D}(\boldsymbol{f} * \boldsymbol{s}_j)]_k\end{aligned}$$

$$\begin{aligned}[\boldsymbol{c}_{v;j}]_k &= \sum_n g_{v;k-n}[\boldsymbol{s}_j]_n \\ &= [\boldsymbol{g}_v * \boldsymbol{s}_j]_k\end{aligned}$$

where $[\boldsymbol{s}_0]_k = s_k$.

Following the approach presented in Section 3.2 let $\phi(t)$ be defined as in Equation 3.25 and

$$\psi'_v(t) = \sum_k \overline{g}_{v;k}\phi(t+k) = \sum_k g^{\dagger}_{v;k}\phi(t-k).$$

Using the initialization given in Equation 3.30 one can show that the coefficients on the right-hand side of Equation 3.32 are the exact discretized multivoice decimated wavelet "series" coefficients $(\mathsf{C}_{\psi'_v}s)_{j,k}$. Furthermore if $\boldsymbol{f}$ is à trous the result $[\boldsymbol{s}]_n = s(n)$ still holds. Similarly, using the initialization given in Equation 3.31 one can show that the coefficients on the right-hand side of Equation 3.32 are the exact multivoice decimated wavelet "series" coefficients $(\mathsf{C}_{\psi'_v}s)_{j,k}$. Again if $\boldsymbol{f}$ is à trous the result $\psi'_v(n) = \psi_v(n)$ still holds.

Extending the above results to the undecimated case can be easily accomplished by using Equation 3.22, that is

$$[\boldsymbol{r}_{v;j}(\boldsymbol{s})]_k = [\boldsymbol{c}_{v;j}(T_{-k}\boldsymbol{s})]_0,$$

where the coefficients $[\boldsymbol{r}_{v;j}]_k = [\boldsymbol{r}_{v;j}(\boldsymbol{s})]_k$ can be computed iteratively over j as follows

$$[\boldsymbol{s}_{j+1}]_k = [(\boldsymbol{U}^j\boldsymbol{f}) * \boldsymbol{s}_j)]_k$$

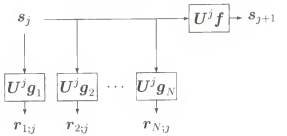


Figure 3.3: Block diagram of one stage of the multivoice discrete undecimated wavelet transform.

$$[\boldsymbol{r}_{v;j}]_k \;=\; [(\boldsymbol{U}^j \boldsymbol{g}_v) * \boldsymbol{s}_j]_k.$$

A block diagram of one stage of the transform is shown in Figure 3.3. One can show that $(\mathsf{r}_{\psi_v} s)_{j,k}$ and $(\mathsf{R}_{\psi_v} s)_{j,k}$ can be computed using the above algorithm by choosing the initializations in Equations 3.30 and 3.31, respectively.

3.4 The Inverse Discrete Wavelet Transform

Although the invertibility of an orthogonal or biorthogonal wavelet transform is relatively straight forward, the same cannot be said with regards to a nonorthogonal wavelet transform. The standard inversion procedure for a nonorthogonal wavelet transform is given by Equation 2.24. However, in order to use this procedure in practice the following approximations are involved [Dau90, Dau92, She96]: (1) The wavelet coefficients themselves are approximated, (2) The duals are replaced by the analyzing wavelet, and (3) The infinite sums are replaced by partial (finite) sums. Each of these approximations can be justified as follows:

1. The wavelet coefficients can be approximated by using $\psi'(t)$ as the analyzing wavelet instead of $\psi(t)$ (see Section 3.2). It is not difficult to argue that if $\psi'(t)$ and $\psi(t)$ are "close" then the sequences $\langle s, \psi'_{j,k} \rangle$ and $\langle s, \psi_{j,k} \rangle$ should be "close" as well (i.e., $(\mathsf{C}_\psi s)_{j,k} \approx (\mathsf{C}_{\psi'} s)_{j,k} = [\boldsymbol{c}_j(\boldsymbol{s})]_k$).

2. Replacing the duals by the analyzing wavelet is a valid approximation as long as the analyzing wavelet generates a "snug frame" (see Section 2.2.1).

3. Replacing the infinite sums by partial sums is justified by the time-frequency localization properties of the wavelet transform. It can be shown that if a signal $s(t)$ is essentially localized to some rectangle in the time-frequency plane $[-T,T]\times([-\Omega_1,-\Omega_0]\cup[\Omega_0,\Omega_1])$ then the nonzero wavelet coefficients lie within or close to that rectangle [Dau92].

The above inversion procedure can be written as follows

$$\begin{aligned} s(t) &= \sum_{j=-\infty}^{+\infty}\sum_{k=-\infty}^{+\infty}(\mathsf{C}_\psi s)_{j,k}\tilde{\psi}^{j,k}(t) \\ &\approx K\sum_{j=-\infty}^{+\infty}\sum_{k=-\infty}^{+\infty}c_{j,k}\psi_{j,k}(t) \\ &\approx K\sum_{(j,k)\text{ in }S}c_{j,k}\psi_{j,k}(t) \end{aligned} \tag{3.34}$$

where $S=\left\{(j,k): m(\psi_{j,k})\in[-T,T] \text{ and } \pm m(\hat{\psi}_{j,k})\in([-\Omega_1,-\Omega_0]\cup[\Omega_0,\Omega_1])\right\}$ and K is a constant determined by the dual approximation. In most practical cases $m(\psi_{j,k})=2^j k$ and $m(\hat{\psi}_{j,k})=2^{-j}m(\hat{\psi})$ (see Equations 2.14 and 2.18).

In practice the finest scale is set to 1 and the wavelet coefficients are computed only for $j\geq 0$ by using the initialization in Equation 3.31. Including this restriction in Equation 3.34 yields

$$s(t)\approx K\sum_{j=0}^{L'-1}\sum_{k}c_{j,k}\psi_{j,k}(t) \tag{3.35}$$

where $L'\leq\log_2\left(\frac{m(\hat{\psi})}{\Omega_0}\right)+1$ and the sum in k is implicitly assumed to be restricted to $k\in\left[-\frac{T}{2^j},\frac{T}{2^j}\right]$. Although these approximations are indeed valid the above inversion procedure has several drawbacks. The most noticeable problems are [She96]: (1) It does not provide a good inverse for narrow-band wavelets (loose frame bounds) and (2) While a partial expansion works quite well for many (sufficiently oscillatory) signals, it fails to achieve good accuracy or requires an excessive number of scales for others. An alternative solution that addresses the problems of the above inversion

procedure is presented in [She93]. Unlike Equation 3.35 the solution in [She93] is based on the undecimated wavelet transform coefficients. In this section we review the main results of this work.

Consider L stages of the discrete wavelet transform (DWT). Then, the DWT maps discrete signals $\mathbb{R}(\mathbb{Z})$ into $\mathbb{C}^{L+1}(\mathbb{Z})$

$$\boldsymbol{s} \overset{\text{DWT}}{\rightarrow} \{\boldsymbol{w}_j : j = 0, \ldots, L-1; \boldsymbol{s}_L\}$$

where $(\mathbb{Z})$ indicates a space of functions on the integers and $\boldsymbol{w}_j$ refers to the coefficients of either the decimated DWT, $\boldsymbol{c}_j$, or the undecimated DWT, $\boldsymbol{r}_j$. For the above transform one has that [She93]

1. Under very weak regularity conditions, the norm of $\boldsymbol{s}_L$ goes to zero as L goes to ∞.

2. For the transform to be nonsingular for finite L, one must include the smoothed signal $\boldsymbol{s}_L$ as well as the wavelet coefficients $\boldsymbol{w}_j$.

3. The transform is linear (although not time invariant in the decimated case) and may be represented by a matrix $\boldsymbol{A}$.

4. For $\boldsymbol{f}$ and $\boldsymbol{g}$ finite, the transform is locally finite but the matrix itself is infinite because convolution acts on arbitrarily long signals.

5. A true inverse does not exist because the image of the transformation is a proper subset of $\mathbb{C}^{L+1}(\mathbb{Z})$. However, the transform is generally injective so $\boldsymbol{A}^\dagger\boldsymbol{A}$ is nonsingular, and one may invert objects in the range space $\mathbb{C}^{L+1}(\mathbb{Z})$ by using the pseudo-inverse [She93]

$$P = (\boldsymbol{A}^\dagger\boldsymbol{A})^{-1}\boldsymbol{A}^\dagger.$$

More generally, if $\boldsymbol{B}$ is a matrix such that $\boldsymbol{BA}$ is nonsingular then

$$Q = (\boldsymbol{BA})^{-1}\boldsymbol{B} \tag{3.36}$$

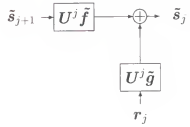


Figure 3.4: Block diagram of one stage of the inverse discrete undecimated wavelet transform.

is also a left inverse. However, the pseudo-inverse is unique to the extent that it is an orthogonal projection of $\mathbb{C}^{L+1}(\mathbb{Z})$ onto the image of the DWT [She93][3]. (Compare this analysis to the discussion on frames in Section 2.2.1.) Although the pseudo-inverse is a natural choice, other issues are involved such as computational complexity, or the the error criterion not being a metric on $\mathbb{C}^{L+1}(\mathbb{Z})$ [She93].

An alternative approach is to invert a single stage of the DWT, thereby inverting the entire transform. For the undecimated DWT, it is sufficient to find filters $\tilde{\boldsymbol{f}}$ and $\tilde{\boldsymbol{g}}$ such that

$$\tilde{\boldsymbol{f}} * \boldsymbol{f} + \tilde{\boldsymbol{g}} * \boldsymbol{g} = \boldsymbol{\delta}. \tag{3.37}$$

For the multivoice case we have

$$\tilde{\boldsymbol{f}} * \boldsymbol{f} + \sum_{v=1}^{N} \tilde{\boldsymbol{g}}_v * \boldsymbol{g}_v = \boldsymbol{\delta}. \tag{3.38}$$

Thus the inverse filter bank of Figure 3.4 (Figure 3.5) is a left inverse for the undecimated DWT of Figure 3.2 (Figure 3.3) provided that Equation 3.37 (Equation 3.38) is satisfied. In fact, as we will see shortly, if one consider a more general class of filters than those satisfying Equation 3.37 (Equation 3.38 in the multivoice case), the inverse filter bank provides a unified framework from which to study various

[3]Note that this holds for non-Euclidean metrics on $\mathbb{C}^{L+1}(\mathbb{Z})$ provided that $\boldsymbol{A}^\dagger$ is replaced by the adjoint transform relative to the metric [She93].

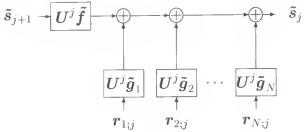


Figure 3.5: Block diagram of one stage of the inverse multivoice discrete undecimated wavelet transform.

inverses. From now on, we will adopt the terminology DWT to refer to the undecimated discrete wavelet transform with or without voices and inverse discrete wavelet transform or DWT^{-1} [She93] to refer to any inverse filter bank whether or not the filters satisfy Equation 3.37 (Equation 3.38 in the multivoice case). Although under this framework the inverse may not be exact, for it to be useful the filters should be chosen to provide an approximate left inverse to the DWT. An exact left inverse is achieved by iterating (see Neumann inverse in Section 3.4.3).

3.4.1 Approximations to the Continuous Inverse

The approach used in [She93] to find suitable inverses is to use the discretization of the continuous inverse in Equation 2.21 as a guideline to provide metrics and appropriate filter normalizations for the DWT^{-1}. Below we include the results for the undecimated wavelet transforms with voices. The decimated wavelet transform with voices is treated as well to draw some useful comparisons.

From Equation 2.21 we have that

$$s(t) = \frac{2}{C_\psi}\int_{-\infty}^{+\infty}\int_0^{+\infty}(\mathcal{W}_\psi s)(a,b)\psi_{a;b}(t)\,\frac{da}{a^2}db. \tag{3.39}$$

In the case of N voices, discretizing the integral for the decimated ($a = 2^j 2^{(v-1)/N}$, $b = 2^j k$) and undecimated ($a = 2^j 2^{(v-1)/N}$, $b = k$) case one obtains [She93]

$$s(t) \approx \frac{2\ln 2}{C_\psi N}\sum_{v=1}^{N}\sum_{j,k}(\mathsf{C}_{\psi_v}s)_{j,k}\frac{1}{2^{(v-1)/N}}\psi_{v;j,k}(t) \tag{3.40}$$

and

$$s(t) \approx \frac{2 \ln 2}{C_\psi N} \sum_{v=1}^{N} \sum_{j,k} (\mathsf{R}_{\psi_v} s)_{j,k} \frac{1}{2^j 2^{(v-1)/N}} \psi^j_{v;k}(t), \tag{3.41}$$

respectively; where

$$\psi_v(t) = \frac{1}{\sqrt{2^{(v-1)/N}}} \psi \left(\frac{t}{2^{(v-1)/N}} \right),$$

$$\psi_{v;j,k}(t) = \frac{1}{\sqrt{2^j}} \psi_v \left(\frac{t}{2^j} - k \right),$$

and

$$\psi^j_{v;k}(t) = \frac{1}{\sqrt{2^j}} \psi_v \left(\frac{t-k}{2^j} \right).$$

Notice that the above normalization for $\psi_v(t)$ is not the same as the one in Section 2.2.3. The normalization used here follows from the sampling of $\psi_{a;b}(t)$ with $a = 2^j 2^{(v-1)/N}$ and $b = 2^j k$ in the decimated case (or $b = k$ in the undecimated case). When discretizing Equation 3.39 one requires the above normalization since the continuous inverse is defined in terms of $\psi_{a;b}(t)$.

By incorporating the square root of $\frac{1}{2^{(v-1)/N}}$ in the above normalization of $\psi_v(t)$ one can eliminate this factor from Equations 3.40 and 3.41. This yields the normalization used in Section 2.2.3. In such a case, expressions equivalent to Equations 3.40 and 3.41 are given by

$$s(t) \approx \frac{2 \ln 2}{C_\psi N} \sum_{v=1}^{N} \sum_{j,k} (\mathsf{C}_{\psi_v} s)_{j,k} \psi_{v;j,k}(t)$$

and

$$s(t) \approx \frac{2 \ln 2}{C_\psi N} \sum_{v=1}^{N} \sum_{j,k} (\mathsf{R}_{\psi_v} s)_{j,k} \frac{1}{2^j} \psi^j_{v;k}(t),$$

respectively. The normalizations used in this section are favored by [She93] because the squares of all four types of wavelet coefficients (decimated/undecimated with/without voices) represent power per Hz.

In order to relate the above approximations to the discrete wavelet transform Shensa [She93] discretizes "Plancherel" formula for wavelets (see Equation 2.11). Restricting $a > 0$ we have that

$$\|s(t)\|_2^2 = \frac{2}{C_\psi} \int_{-\infty}^{+\infty} \int_0^{+\infty} |(\mathcal{W}_\psi s)(a,b)|^2 \, \frac{da}{a^2} db.$$

Discretizing for the decimated and undecimated case with voices one obtains

$$\|s(t)\|_2^2 \approx \frac{2 \ln 2}{C_\psi N} \sum_{v=1}^{N} \sum_{j,k} \frac{1}{2^{(v-1)/N}} \left|(\mathsf{C}_{\psi_v} s)_{j,k}\right|^2. \tag{3.42}$$

and

$$\|s(t)\|_2^2 \approx \frac{2 \ln 2}{C_\psi N} \sum_{v=1}^{N} \sum_{j,k} \frac{1}{2^j 2^{(v-1)/N}} \left|(\mathsf{R}_{\psi_v} s)_{j,k}\right|^2. \tag{3.43}$$

respectively. The goal is to use the above discretization as a guideline for obtaining an expression equivalent to Plancherel's formula for the discrete undecimated wavelet transform. Before proceeding in this direction we provide an interpretation for the additional $\frac{1}{2^j}$ factor appearing in Equations 3.41 and 3.43 when compared with Equations 3.40 and 3.42, respectively. As we will see shortly, this will prove useful in establishing an expression equivalent to Plancherel's formula for the discrete wavelet transform.

It is not difficult to see that for scales[4] coarser than 1 (i.e., $j > 0$) the undecimated wavelet transform has a denser sampling than the decimated wavelet transform. In fact, for any $j > 0$ there are 2^j coefficients in the undecimated wavelet transform for every coefficient in the decimated wavelet transform. Similarly, for scales finer than 1 (i.e., $j < 0$) we have the opposite behavior; for any $j < 0$ there are 2^{-j} coefficients in the decimated wavelet transform for every coefficient in the undecimated wavelet transform. Therefore, the additional $\frac{1}{2^j}$ factor in Equations 3.41 and 3.43 can be "viewed" as balancing the redundancy at scales coarser than

[4] Although $a = 2^j 2^{(v-1)/N}$ is the scale parameter, in this discussion we use "scale" to refer to the term 2^j.

1 and compensating for the conciseness at scales finer than 1. Now consider the undecimated DWT of a discretized signal $\boldsymbol{s}$ (e.g., see Equation 3.31). Assuming the approximation $(\mathsf{R}_{\psi_v} s)_{j,k} \approx [\boldsymbol{r}_{v;j}]_k$ we write the following "equivalent" to Plancherel's formula in Equation 3.43 for the discrete wavelet transform

$$\|\boldsymbol{s}\|^2 \approx \frac{2\ln 2}{C_\psi N} \sum_{v=1}^{N} \sum_{j=0}^{+\infty} \frac{1}{2j2^{(v-1)/N}} \|\boldsymbol{r}_{v;j}\|^2.$$

If there are only L stages (levels) then $\boldsymbol{s}_L$ "replaces" the coefficients $\boldsymbol{r}_{v;j}$ for $j \geq L$ as follows

$$\|\boldsymbol{s}\|^2 \approx \frac{2\ln 2}{C_\psi N} \sum_{v=1}^{N} \sum_{j=0}^{L-1} \frac{1}{2j2^{(v-1)/N}} \|\boldsymbol{r}_{v;j}\|^2 + \frac{1}{2^L} \|\boldsymbol{s}_L\|^2 \tag{3.44}$$

where the factor $\frac{1}{2^L}$ is added to balance the redundancy of the undecimated wavelet transform at the Lth stage (For a different view on the above "derivation" the reader should consult [She93]). When the approximation in Equation 3.44 holds it implies a relationship between the normalizations of the filters $\boldsymbol{f}$ and $\boldsymbol{g}_v$. Consider the output of a one-stage filter bank where the input is an impulse function. Then, Equation 3.44 implies

$$\frac{2\ln 2}{C_\psi N} \sum_{v=1}^{N} \frac{1}{2^{(v-1)/N}} \|\boldsymbol{g}_v\|^2 + \frac{1}{2} \|\boldsymbol{f}\|^2 \approx 1. \tag{3.45}$$

Notice that Equations 3.44 and 3.45 are only meant to be conceptual guidelines and will be used in the next section to define an appropriate expression for the energy of a (nonorthogonal) discrete wavelet transform.

3.4.2 Discrete Inverses

In this section we review the conditions for finite energy and boundedness of the DWT and the DWT^{-1} as the number of stages L becomes infinite. These properties are important to the extent that they reflect directly on the numerical stability of the algorithms themselves [She93]. Although in practice it is not possible to compute an infinite number of stages, the poor numerical behavior of unbounded transforms tends to persist for finite L [She93].

Based on Equations 3.44 and 3.45 one may define the energy of a (nonorthogonal) discrete wavelet transform as follows [She93]: Suppose there exists $c_E > 0$ such that

$$c_E \left(\frac{1}{N} \sum_{v=1}^{N} \frac{\|\boldsymbol{g}_v\|^2}{2^{(v-1)/N}} \right) + \frac{1}{2} \|\boldsymbol{f}\|^2 = 1.$$

Then define the energy by [She93]

$$E = \sum_{j=0}^{L-1} \sum_{v=1}^{N} \frac{1}{2^j 2^{(v-1)/N} N} c_E \|\boldsymbol{r}_{vj}\|^2 + \frac{1}{2^L} \|\boldsymbol{s}_L\|^2. \tag{3.46}$$

This is to be interpreted to mean that $(\sqrt{c_E}\boldsymbol{r}_{vj}, \boldsymbol{s}_L)$ is an energy density for the DWT and the total energy is given by applying the metric given by the $(LN+1)$-dimensional diagonal matrix with elements $\{2^{-j}2^{-(v-1)/N}, 2^{-L}\}$ [She93].

It follows from Equation 3.46 that for $L = \infty$ the discrete wavelet transform is bounded and has a bounded inverse if and only if there exist $0 < A \leq B < \infty$ such that [She93]

$$A\|\boldsymbol{s}\|^2 \leq E(\boldsymbol{r}) \leq B\|\boldsymbol{s}\|^2.$$

The following is a sufficient condition for it to be satisfied in the case of a single voice [She93]

$$0 < A \leq \frac{\frac{1}{2} c_E \left|\tilde{g}\left(e^{j\omega}\right)\right|^2}{1 - \frac{1}{2}\left|\tilde{f}\left(e^{j\omega}\right)\right|^2} \leq B < \infty \text{ for all } \omega. \tag{3.47}$$

At $\omega = 0$, Equation 3.47 is also a necessary condition [She93]. Hence, one must have $\frac{1}{2}\left|\tilde{f}(1)\right|^2 \leq 1$. Since $\tilde{f}(1) = \sqrt{2}$ (see Equation 3.27), the condition in Equation 3.47 requires $\sum_k g_k = 0$ that is, that $\boldsymbol{g}$ be a highpass filter [She93]. (In the case of multiple voices all $\boldsymbol{g}_v$ must be highpass filters [She93].) Boundedness of the DWT^{-1} for arbitrary L also imposes the constraint [She93]

$$\left|\tilde{\tilde{f}}(1)\right| \leq \frac{1}{\sqrt{2}}$$

Similarly, if the composition of the inverse and forward transform is to be bounded for arbitrary L [She93]

$$\left|\tilde{\tilde{f}}(1)\tilde{f}(1)\right| \leq 1.$$

A sufficient condition for the composition of DWT^{-1} and DWT to be a bounded transform may also be derived [She93] and is given by

$$\left|\tilde{\tilde{f}}(e^{j\omega})\tilde{f}(e^{j\omega})\right| + C^{-1}\left|\tilde{\tilde{g}}(e^{j\omega})\tilde{g}(e^{j\omega})\right| \leq 1 \text{ for all } \omega,$$

with $1 \leq C < \infty$. Such a C exists provided $\left|\tilde{\tilde{f}}(e^{j\omega})\tilde{f}(e^{j\omega})\right| \leq 1$ and

$$\frac{\left|\tilde{\tilde{g}}(e^{j\omega})\tilde{g}(e^{j\omega})\right|}{1 - \left|\tilde{\tilde{f}}(e^{j\omega})\tilde{f}(e^{j\omega})\right|} \leq C \text{ for all } \omega.$$

Although it would be desirable that the DWT^{-1} be an exact left inverse (see Equations 3.37 and 3.38) in practice this leads to filters that are too long for most applications [She93]. As mentioned before, an alternative approach is to provide an approximate left inverse to the DWT and refine it by iterating (see Section 3.4.3). In [She93], three filter pairs that satisfy the above necessary conditions and provide useful approximations are studied. Here we consider the filter pairs that are relevant to our study: (1) The double integral type filters $\tilde{\boldsymbol{f}} = \boldsymbol{\delta}/\sqrt{2}$ and $\tilde{\boldsymbol{g}} = c\boldsymbol{g}^{\dagger}$, and (2) The adjoint filters $\tilde{\boldsymbol{f}} = \boldsymbol{f}^{\dagger}/2$ and $\tilde{\boldsymbol{g}} = c\boldsymbol{g}^{\dagger}$. The filter pair not considered here can be used only if $\psi(t)$ is analytic or $\hat{\psi}(\omega)$ is real. The reader can verify that the wavelet used in this study (see Chapter 4) does not satisfy any of the above conditions.

By considering the output of the DWT^{-1} in the case of a single voice [She93]

$$\tilde{\boldsymbol{s}} = \sum_{j=0}^{L-1} \prod_{i=0}^{j-1} \left[\left(\boldsymbol{U}^i \tilde{\boldsymbol{f}}\right) *\right] \left(\boldsymbol{U}^j \tilde{\boldsymbol{g}}\right) * \boldsymbol{r}_j + \prod_{j=0}^{L-1} \left[\left(\boldsymbol{U}^j \tilde{\boldsymbol{f}}\right) *\right] \boldsymbol{s}_L \tag{3.48}$$

one may interpret the above filters as approximations to the continuous inverse formula in Equation 3.39. Furthermore, for the adjoint filters it can be shown that the DWT^{-1} is indeed the adjoint of the DWT under the energy metric in Equation 3.46

[She93]. Below we include the interpretation for the double integral type filters. For the adjoint filters consult [She93].

For the double integral type filters one can show that [She93]

$$\begin{aligned}
[\tilde{\boldsymbol{s}}]_0 &= c\sum_{j=0}^{L-1}\frac{1}{\sqrt{2^j}}\left[\left(\boldsymbol{U}^j\boldsymbol{g}^\dagger\right)*\boldsymbol{r}_j\right]_0+\mathrm{DC}\\
&= c\sum_{j=0}^{L-1}\frac{1}{\sqrt{2^j}}\sum_{\boldsymbol{k}}[\boldsymbol{r}_j]_{2^j\boldsymbol{k}}\,\overline{g}_{\boldsymbol{k}}+\mathrm{DC}\\
&= c\sum_{j=0}^{L-1}\sum_{\boldsymbol{k}}[\boldsymbol{c}_j]_{\boldsymbol{k}}\,\frac{1}{\sqrt{2^j}}\psi(-k)+\mathrm{DC}\\
&= c\sum_{j=0}^{L-1}\sum_{\boldsymbol{k}}c_{j,\boldsymbol{k}}\psi_{j,\boldsymbol{k}}(0)+\mathrm{DC} \qquad (3.49)
\end{aligned}$$

where DC refers to the last term of Equation 3.48. This "coincides" up to the DC term with the standard approximation in Equation 3.35 at time $t=0$. By incorporating the dependency on $\boldsymbol{s}$ in Equation 3.49 we have that

$$[\tilde{\boldsymbol{s}}]_0 = c\sum_{j=0}^{L-1}\sum_{\boldsymbol{k}}[\boldsymbol{c}_j(\boldsymbol{s})]_{\boldsymbol{k}}\psi_{j,\boldsymbol{k}}(0)+\mathrm{DC}.$$

Since the undecimated transform and its inverse are time invariant one may write [She93]

$$[\tilde{\boldsymbol{s}}]_n = c\sum_{j=0}^{L-1}\sum_{\boldsymbol{k}}[\boldsymbol{c}_j(T_{-n}\boldsymbol{s})]_{\boldsymbol{k}}\psi_{j,\boldsymbol{k}}(0)+\mathrm{DC}. \qquad (3.50)$$

It is not difficult to show that the above equation represents a discrete approximation to the double integral formula in Equation 3.39. Consider

$$\begin{aligned}
s(t+\tau)=(T_{-\tau}s)(t) &= \frac{2}{C_\psi}\int_{-\infty}^{\infty}\int_0^{\infty}(\mathcal{W}_\psi(T_{-\tau}s))(a,b)\psi_{a;b}(t)\,\frac{da}{a^2}db\\
&= \frac{2}{C_\psi}\int_{-\infty}^{\infty}\int_0^{\infty}(\mathcal{W}_\psi s)(a,b+\tau)\psi_{a;b}(t)\,\frac{da}{a^2}db.
\end{aligned}$$

At $t=0$ this yields

$$s(\tau)=\frac{2}{C_\psi}\int_{-\infty}^{\infty}\int_0^{\infty}(\mathcal{W}_\psi s)(a,b+\tau)\psi_{a;b}(0)\,\frac{da}{a^2}db.$$

Discretizing (see the discussion about the restrictions on j and k in relation to Equation 3.35)

$$s(n) \approx \frac{2\ln 2}{C_\psi} \sum_{j=0}^{L'-1} \sum_k (\mathcal{W}_\psi s)(2^j, 2^j k + n)\psi_{j,k}(0).$$

But $(\mathcal{W}_\psi s)(2^j, 2^j k+n) = (\mathcal{W}_\psi (T_{-n} s))(2^j, 2^j k) = (\mathsf{C}_\psi (T_{-n} s))_{j,k} \approx [\boldsymbol{c}_j (T_{-n}\boldsymbol{s})]_k$. Therefore,

$$s(n) \approx \frac{2\ln 2}{C_\psi} \sum_{j=0}^{L'-1} \sum_k [\boldsymbol{c}_j (T_{-n}\boldsymbol{s})]_k \psi_{j,k}(0). \tag{3.51}$$

This "coincides" up to the DC term with Equation 3.50. Notice that Equation 3.50 should be a better approximation since the DC term replaces the energy lost in truncating the expansion at $j = L - 1$.

The above formulation is extended to N voices by considering the following filters [She93]: (1) The double integral type filters

$$\tilde{\boldsymbol{f}} = \boldsymbol{\delta}/\sqrt{2} \text{ and } \tilde{\boldsymbol{g}}_v = \frac{c}{2^{(v-1)/N}} \boldsymbol{g}_v^\dagger, \tag{3.52}$$

and (2) The adjoint filters

$$\tilde{\boldsymbol{f}} = \boldsymbol{f}^\dagger/2 \text{ and } \tilde{\boldsymbol{g}}_v = \frac{c}{2^{(v-1)/N}} \boldsymbol{g}_v^\dagger. \tag{3.53}$$

For additional details the reader should consult [She93].

Finally, we briefly review the selection of the parameter c for the above filters. Although there are several approaches proposed in [She93] we only review the most successful ones here: (1) Setting the energy of the impulse response of a single stage of the composition of DWT^{-1} and DWT to be equal to one, i.e.,

$$\| c \sum_{v=1}^{N} \frac{1}{2^{(v-1)/N}} \boldsymbol{g}_v^\dagger * \boldsymbol{g}_v + \tilde{\boldsymbol{f}} * \boldsymbol{f} \|^2 = 1, \tag{3.54}$$

and (2) Setting

$$c = \frac{2\ln 2}{C_\psi N} \tag{3.55}$$

by analogy to the continuous case. (Compare Equations 3.50 and 3.51. The normalization $1/N$ comes from the generalization to multiple voices.)

3.4.3 Neumann Inverse

Given a discrete wavelet transform $\boldsymbol{A}$ and an approximate inverse $\boldsymbol{C}$, an exact left inverse (see Equation 3.36) can be computed by using the Neumann inverse for linear operators. Here, $\boldsymbol{A}$ and $\boldsymbol{C}$ are the transforms DWT and DWT^{-1}, respectively. Let $\mathcal{S}$ be a linear operator in a Hilbert space then the Neumann inverse is given by

$$\mathcal{S}^{-1} = \mu \sum_{k=0}^{\infty} (\mathcal{I} - \mu \mathcal{S})^k, \tag{3.56}$$

where $\|\mathcal{I} - \mu\mathcal{S}\| < 1$ to insure convergence [Dau92, She93]. (Notice this procedure is also used to compute the duals in Equation 2.32.) Equation 3.56 implies that the inverse of $\boldsymbol{CA}$ is given by

$$(\boldsymbol{CA})^{-1} = \mu \sum_{k=0}^{\infty} (\boldsymbol{I} - \mu \boldsymbol{CA})^k.$$

Therefore, given a signal $\boldsymbol{s}$ we have that

$$\boldsymbol{s} = \mu \sum_{k=0}^{\infty} (\boldsymbol{I} - \mu \boldsymbol{CA})^k \boldsymbol{CA}\boldsymbol{s}.$$

Truncating after $N+1$ terms we have that

$$\boldsymbol{s}^N = \mu \sum_{k=0}^{N} (\boldsymbol{I} - \mu \boldsymbol{CA})^k \boldsymbol{CA}\boldsymbol{s},$$

which can be computed iteratively as follows

Algorithm 3.4.1

$k \leftarrow 0$, $\boldsymbol{e}^k \leftarrow \boldsymbol{x}^k \leftarrow \mathrm{DWT}^{-1} \circ \mathrm{DWT}\boldsymbol{s}$ {Decomposition followed by reconstruction.}

while $k < N$ **do**

$k \leftarrow k + 1$

$\boldsymbol{e}^k \leftarrow \boldsymbol{e}^{k-1} - \mu\, \mathrm{DWT}^{-1} \circ \mathrm{DWT}\boldsymbol{e}^{k-1}$

$\boldsymbol{x}^k \leftarrow \boldsymbol{x}^{k-1} + \boldsymbol{e}^k$

end while

$\boldsymbol{s}^k \leftarrow \mu \boldsymbol{x}^k$ {kth left inverse.}

As pointed out in [She93], Algorithm 3.4.1 suffers from a computational drawback as there is a huge increase in the support of the signal under the mapping $\mathrm{DWT}^{-1} \circ \mathrm{DWT}$. In [She93], a heuristic solution that consists in clipping the support of the iterations to that of the DWT of $\boldsymbol{s}$ is proposed. As we will see in Chapter 5 we avoid this problem by restricting $\boldsymbol{s}$ to periodically extended signals.

CHAPTER 4
THE 1–D SINE–GABOR WAVELET

In this chapter we introduce a new wavelet: the sine-Gabor function. We show that this function not only satisfies the wavelet admissibility condition but under certain conditions achieves nearly optimum time-frequency localization and generates a frame of wavelets. Furthermore, we show that there is a trade-off between the tightness of the frame and the time-frequency localization properties of the sine-Gabor wavelet. We demonstrate how this trade off can be overcome by introducing voices.

4.1 The Sine–Gabor Wavelet

It is well known that the modulated Gaussian or Gabor function is the only function that achieves optimum time-frequency localization. The question arises as to whether the Gabor function,

$$\psi_G(t) = K_G\, e^{-t^2/2\sigma_0^2} e^{-j\omega_0 t},$$

can be used in the context of wavelet analysis and synthesis. Unfortunately, this is not possible because the Gabor function is not an admissible wavelet (for details on the admissibility condition for wavelets see Section 2.2.) Therefore, optimum time-frequency localization via the wavelet transform is not possible. However, a modified version of the Gabor function, know as the Morlet wavelet [Dau92],

$$\psi_M(t) = K_M\, e^{-t^2/2\sigma_0^2} \left(e^{-j\omega_0 t} - e^{-\sigma_0^2 \omega_0^2/2} \right),$$

satisfies the wavelet admissibility condition, and for $\sigma_0 = 1$, $\omega_0 = \pi\,(2/\ln 2)^{1/2}$, achieves nearly optimum time-frequency localization and generates a frame of

wavelets (for more details about wavelet frames see Section 2.2 and 2.2.1.) In fact, the error incurred in approximating the Morlet wavelet with a Gabor function under these conditions is negligible [Dau92]. Although Gabor functions are not wavelets in the strict sense, the term "Gabor wavelet" is used in the literature (e.g., [Bov92]). We will refer to Gabor functions as Gabor wavelets provided that $e^{-\sigma_0^2\omega_0^2/2}$ is negligible.

Having a frame of wavelets, $\psi_{j,k}(t)$, $j,k \in \mathbb{Z}$, is a desirable property because it allows to completely characterize and reconstruct a signal $s(t)$ from its discrete wavelet coefficients $\langle s(t), \psi_{j,k}(t)\rangle$ (see Section 2.2.) In practice it is desirable to have a frame of wavelets for which both analysis and synthesis can be computed efficiently.

We show that the sine-Gabor function (imaginary part of the Gabor function) achieves nearly optimum time-frequency localization and at the same time generates a frame of wavelets. Furthermore, we show that both analysis and synthesis using the sine-Gabor wavelet can be computed efficiently.

The fact that the linear combination of two wavelets is also a wavelet gives us an easy way to show that the sine-Gabor function is a wavelet. By taking the real and imaginary parts of the Morlet wavelet we have that

$$\operatorname{Re}\psi_M(t) = K_M e^{-x^2/2\sigma_0^2}\left(\cos(\omega_0 t) - e^{-\sigma_0^2\omega_0^2/2}\right),$$

and

$$\operatorname{Im}\psi_M(t) = -K_M e^{-x^2/2\sigma_0^2}\sin(\omega_0 t).$$

It follows that the sine-Gabor function is indeed a wavelet. It also follows that the cosine-Gabor function (real part of the Gabor function) is not a wavelet for otherwise the Gabor function would be a wavelet.

The question as to whether or not the sine-Gabor wavelet generates a frame of wavelets and the efficient computation of both analysis and synthesis using the sine-Gabor wavelet are addressed below.

4.2 The Sine–Gabor Wavelet Frame

We showed in Section 4.1 that sine-Gabor functions are wavelets. In this section we study the time-frequency localization characteristics of the sine-Gabor wavelets and also establish that under certain conditions they generate wavelet frames.

The sine-Gabor wavelet is given by

$$\psi(t) = K_\psi e^{-x^2/2\sigma_0^2} \sin(\omega_0 t), \tag{4.1}$$

where

$$K_\psi = \pi^{-1/4} \left(\frac{2}{\sigma_0 (1 - e^{-\sigma_0^2 \omega_0^2})} \right)^{1/2}$$

was chosen so that $\|\psi\|_2 = 1$. The time and frequency resolution of the sine-Gabor wavelet are given by the root mean square extent of the function and its Fourier transform, respectively (see Equations 2.1 in Section 2.1.) It follows from Equations 4.1 and 2.1 that the time-resolution of the sine-Gabor wavelet is given by

$$\sigma(\psi) = \left(\frac{\sigma_0^2 (1 - e^{-\sigma_0^2 \omega_0^2} (1 - 2\sigma_0^2 \omega_0^2))}{2(1 - e^{-\sigma_0^2 \omega_0^2})} \right)^{1/2}. \tag{4.2}$$

The Fourier transform of the sine-Gabor wavelet is given by

$$\hat{\psi}(\omega) = K_{\hat{\psi}} \frac{e^{-\sigma_0^2 (\omega - \omega_0)^2/2} - e^{-\sigma_0^2 (\omega + \omega_0)^2/2}}{2j},$$

where

$$K_{\hat{\psi}} = K_\psi \sqrt{2\pi \sigma_0}.$$

Computing the root mean square extent of $\hat{\psi}(\omega)$ as defined in Equation 2.1 would lead to misleading results. It can be verified that if a function $\psi(t)$ satisfies the wavelet admissibility condition and is also a window function then its Fourier transform $\hat{\psi}(\omega)$ must vanish at the origin (see Equation 2.16 in Section 2.2.) In addition if

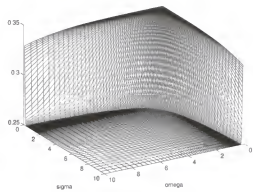


Figure 4.1: Square of the product of the time and frequency resolution for the sine-Gabor wavelet as a function of σ_0 and ω_0.

the wavelet $\psi(t)$ is real we have that $\hat{\psi}(\omega) = \overline{\hat{\psi}(-\omega)}$, so that $|\hat{\psi}(\omega)|$ is a symmetric function around the origin. It is therefore more suitable to restrict our attention to positive frequencies for this class of wavelets and use

$$\sigma(\hat{\psi}) = \left\{ \frac{2}{\|\hat{\psi}\|_2^2} \int_0^\infty (\omega - m(\hat{\psi}))^2 |\hat{\psi}(\omega)|^2 \, d\omega \right\}^{1/2},$$

for the frequency-resolution, where

$$m(\hat{\psi}) = \frac{2}{\|\hat{\psi}\|_2^2} \int_0^\infty \omega |\hat{\psi}(\omega)|^2 d\omega.$$

It is easy to verify that the sine-Gabor wavelet satisfies all the conditions mentioned above and that its frequency-resolution is given by

$$\sigma(\hat{\psi}) = \left(\frac{1 + 2\sigma_0^2\omega_0^2 - e^{-\sigma_0^2\omega_0^2}}{2\sigma_0^2(1 - e^{-\sigma_0^2\omega_0^2})} - (m(\hat{\psi}))^2 \right)^{1/2} \tag{4.3}$$

where

$$m(\hat{\psi}) = \frac{\omega_0 \, \mathrm{erf}(\sigma_0\omega_0)}{1 - e^{-\sigma_0^2\omega_0^2}}. \tag{4.4}$$

Figure 4.1 shows the square of the product of the time and frequency resolution for the sine-Gabor wavelet as a function of σ_0 and ω_0. For $\sigma_0\omega_0 \geq 5/2$, the sine-Gabor wavelet is within 1% of the optimum time-frequency localization lower bound given by the uncertainty principle, $\sigma^2(\psi)\,\sigma^2(\hat{\psi}) \geq 1/4$ (for more details see Equation 2.8 in Section 2.1.)

Next, we establish that the sine-Gabor function does indeed generate a frame of wavelets. General conditions on a wavelet $\psi(t)$ under which a frame is obtained and estimates for the frame bounds are discussed in Section 2.2.2. A wavelet $\psi(t)$ generates a frame of wavelets if the following condition is satisfied

$$|\hat{\psi}(\omega)| \leq C|\omega|^{\alpha}(1+|\omega|)^{-\beta}, \tag{4.5}$$

with $\alpha > 0$, $\beta > \alpha + 1$.

The magnitude of the Fourier transform of the sine-Gabor wavelet may be written as follows

$$\begin{aligned} |\hat{\psi}(\omega)| &= \frac{K_{\hat{\psi}}}{2}\left(1 - e^{-2\sigma_0^2\omega_0|\omega|}\right)e^{-\sigma_0^2(|\omega|-\omega_0)^2/2} \\ &\leq \frac{K_{\hat{\psi}}}{2}(2\sigma_0^2\omega_0|\omega|)^{1/2}e^{-\sigma_0^2(|\omega|-\omega_0)^2/2} \\ &\leq \frac{K_{\hat{\psi}}}{2}\frac{(2\sigma_0^2\omega_0|\omega|)^{1/2}}{1+\sigma_0^2(|\omega|-\omega_0)^2/2}, \end{aligned}$$

where the first and second inequalities are established by using $1 - e^{-\omega} \leq \omega^{\gamma}$, $\forall\, \omega > 0$, $0 \leq \gamma \leq 1$, and $e^{-\omega^2/2} \leq \frac{1}{1+\omega^2/2}$, $\forall\, \omega$, respectively. It follows that the sine-Gabor wavelet satisfies Equation 4.5 with $\alpha = 0.5$ and $\beta = 1.85$ for some C sufficiently large.

4.2.1 Frame Bounds

Table 4.1 and Figure 4.2 show the estimated frame bounds and sine-Gabor wavelet for various values of σ_0 and ω_0. The results in Table 4.1(a) show that it is possible to obtain a frame of wavelets by using the sine-Gabor wavelet and at the same time achieve nearly optimum time-frequency localization, $\sigma^2(\psi)\sigma^2(\hat{\psi}) = 0.2525$. Table 4.1(b) shows that by trading off time-frequency localization, $\sigma^2(\psi)\sigma^2(\hat{\psi}) = 0.3297$, the frame bounds can be made tighter. As mentioned in Section 2.2, snug frame bounds are a desirable property because the dual wavelet can be approximated by $\frac{2}{A+B}\psi(t)$. The error between the original signal and its reconstruction using the above approximation is given by $\frac{r}{2+r}||f||_2$, where $r = B/A - 1$ (see Section 2.2.1 for more details.) For the sine-Gabor wavelet with $\sigma_0 = 1$, $\omega_0 = 1$, the error is 2.55% for

Table 4.1: Frame bounds for the sine-Gabor wavelet with (a) $\sigma_0 = 5/2$, $\omega_0 = 1$, (b) $\sigma_0 = 1$, $\omega_0 = 1$, and (c) $\sigma_0 = 1.0657$, $\omega_0 = 0.0299$. The elementary dilation parameter $a_0 = 2$ in all cases.

(a) Frame bounds with $\sigma_0 = 5/2$, $\omega_0 = 1$.

b_0	A	B	B/A
1.00	4.3873	5.6329	1.2839
1.25	3.5099	4.5063	1.2839
1.50	2.9216	3.7586	1.2865
1.75	2.4089	3.3169	1.3769
2.00	1.6016	3.4085	2.1281
2.25	0.3963	4.0572	10.2385

(b) Frame bounds with $\sigma_0 = 1$, $\omega_0 = 1$.

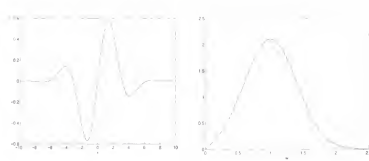
b_0	A	B	B/A
0.50	8.5076	8.6988	1.0225
0.75	5.6715	5.7995	1.0226
1.00	4.1918	4.4114	1.0524
1.25	2.8464	4.0362	1.4180
1.50	1.2635	4.4720	3.5395

(c) Frame bounds with $\sigma_0 = 1.0657$, $\omega_0 = 0.0299$.

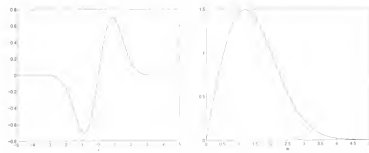
b_0	A	B	B/A
0.75	7.2030	7.3285	1.0174
1.00	5.3999	5.4988	1.0183
1.25	4.2456	4.4733	1.0536
1.50	3.1604	4.1054	1.2990
1.75	1.9203	4.3075	2.2432
2.00	0.6524	4.7969	7.3523

Table 4.2: Frame bounds for the first derivative of a Gaussian with $\sigma_0 = 1.0657$.

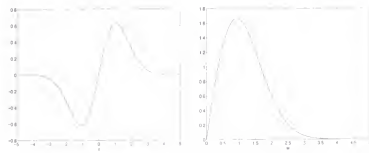
b_0	A	B	B/A
0.75	7.2043	7.3298	1.0174
1.00	5.4008	5.4997	1.0183
1.25	4.2465	4.4739	1.0536
1.50	3.1615	4.1055	1.2986
1.75	1.9218	4.3071	2.2412
2.00	0.6541	4.7962	7.3328



(a)



(b)



(c)

Figure 4.2: Sine-Gabor wavelet (left) and magnitude of its Fourier transform (right) with (a) $\sigma_0 = 5/2$, $\omega_0 = 1$, (b) $\sigma_0 = 1$, $\omega_0 = 1$ and (c) $\sigma_0 = 1.0657$, $\omega_0 = 0.0299$.

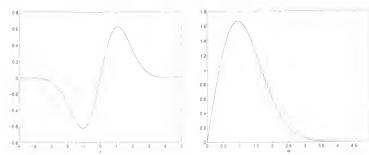


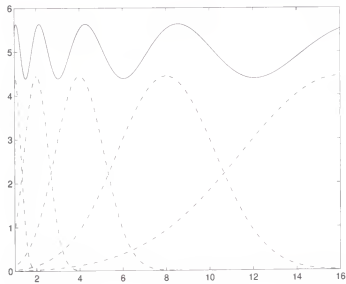
Figure 4.3: First derivative of a Gaussian (left) and the magnitude of its Fourier transform (right) with $\sigma_0 = 1.0657$.

$a_0 = 2, b_0 = 1$. The parameters for the sine-Gabor wavelet used in Table 4.1(c) were obtained by using a Simplex search method to find the minimum of the difference between B and A as a function of σ_0 and ω_0. The error in this case is 0.90% for $a_0 = 2, b_0 = 1$. This reduced error is at the expense of time-frequency localization, $\sigma^2(\psi)\sigma^2(\dot{\psi}) = 0.3401$. An important observation follows from this last example: For $\sigma_0 = 1.0657$, $\omega_0 = 0.0299$ the sine-Gabor wavelet is nearly equal[1] (up to a sign) to the first derivative of a Gaussian with $\sigma_0 = 1.0657$. Therefore, the first derivative of a Gaussian (Canny's approximation to the optimal step edge detector [Can86]) can be used to completely characterize and reconstruct a signal from its discrete wavelet coefficients. Figure 4.3 shows the first derivative of a Gaussian and the magnitude of its Fourier transform. Table 4.2 shows the frame bounds for the first derivative of a Gaussian with $\sigma_0 = 1.0657$. The advantage of the sine-Gabor function over the first derivative of a Gaussian is that by changing the product of $\sigma_0\omega_0$, one is able to trade-off between Canny's criteria for designing optimal step edge detectors, that is, the product of signal-to-noise ratio and localization, $\Sigma\Delta$, and the multiple response constraint, r [Meh92]. This is a property of Canny's optimal step edge detector that is not possible to achieve with the first derivative of a Gaussian. Table 4.3 shows the frame bounds for the sine-Gabor wavelet for $0.75 \leq \sigma_0\omega_0 \leq 3.75$. Figure 4.4 shows the frequency coverage of Equation 2.34 for the sine-Gabor wavelet. Notice that this equation is in fact the dominant term in the computation of the frame bounds in this case (compare Figure 4.4 and Table 4.1) and that by trading off the bandwidth of the wavelet one is able to achieve a tighter frame.

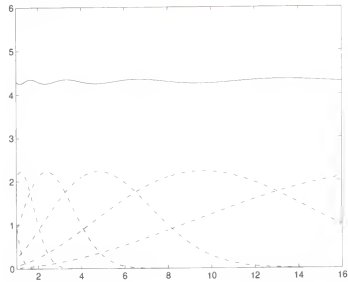
4.2.2 Multivoice Frame Bounds

In the previous section we showed that the sine-Gabor wavelet does indeed generate a wavelet frame but that it is not possible to generate a snug sine-Gabor

[1] It is assumed that both functions have the same L^2-norm.



(a)



(b)

Figure 4.4: Frequency coverage of $\sum_{j=-\infty}^{+\infty} \left|\hat{\psi}(2^j\omega)\right|^2$ for the sine-Gabor wavelet over four octaves (solid) and significant terms $\left|\hat{\psi}(2^j\omega)\right|^2$ (dashed) contributing to the sum over the range shown, (a) $\sigma_0 = 5/2$, $\omega_0 = 1$, (b) $\sigma_0 = 1$, $\omega_0 = 1$.

Table 4.3: Frame bounds for the sine-Gabor wavelet for $0.75 \leq \sigma_0\omega_0 \leq 3.75$. The elementary dilation parameter $a_0 = 2$ and the elementary translation step $b_0 = 1$ in all cases.

σ_0	A	B	B/A
3.75	2.5933	6.8524	2.6423
3.50	2.9659	6.5120	2.1956
3.25	3.3385	6.2107	1.8603
3.00	3.7031	5.9576	1.6088
2.75	4.0546	5.7629	1.4213
2.50	4.3873	5.6329	1.2839
2.25	4.6920	5.5627	1.1856
2.00	4.9418	5.5248	1.1180
1.75	5.0866	5.4618	1.0738
1.50	5.0580	5.2931	1.0465
1.25	4.7875	4.9433	1.0325
1.00	4.1918	4.4114	1.0524
0.75	2.8851	4.0837	1.4154

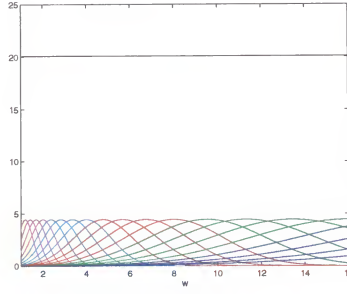


Figure 4.5: Frequency coverage of $\sum_{v=1}^{N} \sum_{j=-\infty}^{+\infty} \left|\hat{\psi}_v(2^j\omega)\right|^2$ for the multivoice sine-Gabor wavelet over four octaves (solid) and significant terms $\left|\hat{\psi}_v(2^j\omega)\right|^2$ (colored) contributing to the sum over the range shown for $N = 4$, $\sigma_0 = 5/2$, $\omega_0 = 1$. Colors are used to denote the four voices within each octave.

Table 4.4: Frame bounds for the sine-Gabor wavelet with $\sigma_0 = 5/2$, $\omega_0 = 1$, (a) $N = 2$, (b) $N = 3$, and (c) $N = 4$. The elementary dilation parameter $a_0 = 2$ in all cases.

(a) Frame bounds with $N = 2$.

b_0	A	B	B/A
0.25	40.0790	40.1998	1.0030
0.50	20.0395	20.0999	1.0030
0.75	13.3597	13.3999	1.0030
1.00	10.0197	10.0500	1.0030
1.25	8.0158	8.0400	1.0030
1.50	6.6765	6.7033	1.0040
1.75	5.6274	5.8410	1.0379
2.00	4.4175	5.6174	1.2716
2.25	2.8891	6.0308	2.0874

(b) Frame bounds with $N = 3$.

b_0	A	B	B/A
0.25	60.2085	60.2097	1.0000
0.50	30.1042	30.1049	1.0000
0.75	20.0695	20.0699	1.0000
1.00	15.0521	15.0524	1.0000
1.25	12.0417	12.0420	1.0000
1.50	10.0314	10.0383	1.0007
1.75	8.5028	8.6998	1.0232
2.00	6.9229	8.1294	1.1743
2.25	5.0273	8.3525	1.6614

(c) Frame bounds with $N = 4$.

b_0	A	B	B/A
0.25	80.2788	80.2788	1.0000
0.50	40.1394	40.1394	1.0000
0.75	26.7596	26.7596	1.0000
1.00	20.0697	20.0697	1.0000
1.25	16.0558	16.0558	1.0000
1.50	13.3765	13.3831	1.0005
1.75	11.3686	11.5682	1.0175
2.00	9.4022	10.6675	1.1346
2.25	7.0938	10.7459	1.5148

wavelet frame and at the same time obtain nearly optimum time-frequency localization. This limitation can be overcome by the introduction of voices (see Section 2.2.3). As mentioned before voices not only aid in obtaining a tighter frame but do so without giving up too much freedom in choosing the wavelet or its bandwidth. In addition, voices or suboctave sampling is a desirable property for applications in the area of signal analysis and time-frequency localization [She96]. Table 4.4 shows the frame bounds for the multivoice sine-Gabor wavelet using the estimates described in Section 2.2.3 and fractionally dilated versions of a single wavelet as described in Equation 2.35. Notice that with two voices ($N = 2$) the error between the original signal and its reconstruction using $\frac{2}{A+B}\psi(t)$ as the dual wavelet is 0.15% for $a_0 = 2$, $b_0 = 1$ (see Section 4.2.1 for a discussion on the error.) This is achieved while preserving nearly optimum time-frequency localization $\sigma^2(\psi)\,\sigma^2(\hat{\psi}) = 0.2525$ (within 1% of the optimum). For $N = 3$ and $N = 4$ the error is negligible for $a_0 = 2$, $b_0 = 1$. In contrast, the error with $N = 1$ is 12.43% for $a_0 = 2$, $b_0 = 1$ and the same time-frequency resolution (see Table 4.1(a)). Figure 4.5 shows the frequency cover of Equation 2.36 for the sine-Gabor wavelet with $N = 4$.

CHAPTER 5
THE 1-D DISCRETE SINE-GABOR WAVELET TRANSFORM

In this chapter we show how to use and extend the approach introduced in Chapter 3 to compute the forward and inverse discrete wavelet transform using the sine-Gabor function as the analyzing wavelet. We also evaluate the quality of the approximating wavelet $\psi'(t)$ for the sine-Gabor wavelet.

5.1 À Trous Filters and Wavelets

Before turning into the computation of the discrete wavelet transform using the sine-Gabor function as the analyzing wavelet, we evaluate the quality of the resulting approximation $\psi'(t)$ for the sine-Gabor wavelet given a suitable à trous filter (see Equations 3.8, 3.29 and 4.1). Given an à trous filter, the Fourier transform of the corresponding scaling function is given by Equation 3.25. It is assumed that suitable regularity conditions are imposed so that the inverse Fourier transform of the product in Equation 3.25 converges to a reasonably behaved function (for more details see [Dau88, Mal89b, She92]). The wavelet $\psi'(t)$ is then given by combining Equations 3.26 and 3.29, where g_k are the samples of the sine-Gabor wavelet $\psi(t)$ in Equation 4.1, i.e., $g_k = \overline{\psi(-k)}$. Unfortunately, it is not possible to compute either the scaling function or the wavelet $\psi'(t)$ as this requires an infinite number of iterations. In order to be able to evaluate the quality of the approximating wavelet we instead compute a finite number of iterations and compare the samples of the approximating wavelet to the exact samples of the sine-Gabor wavelet. Based on Equation 3.26 we define the jth iteration of the scaling function $\phi(t)$ as

$$\phi_{(j)}(t) = \sum_k [(\overline{\boldsymbol{DF}})^j]_{0,k} \sqrt{2^j} \chi(2^j t - k). \tag{5.1}$$

It follows that

$$\phi(t) = \lim_{j\to\infty} \phi_{(j)}(t).$$

Based on Equations 3.29 and 5.1 and with the aid of Equation 3.6 we define the jth iteration of the approximating wavelet $\psi'(t)$ as

$$\psi'_{(j)}(t) = \sum_k [\overline{\boldsymbol{G}}(\overline{\boldsymbol{DF}})^j]_{0,k} \sqrt{2^j} \chi(2^j t - k). \tag{5.2}$$

It follows that

$$\psi'(t) = \lim_{j\to\infty} \psi'_{(j)}(t).$$

From Equations 5.1 and 5.2 it follows that

$$\phi_{(j)}\left(\frac{n}{2^j}\right) = \sqrt{2^j}[(\overline{\boldsymbol{DF}})^j]_{0,n} \tag{5.3}$$

and

$$\psi'_{(j)}\left(\frac{n}{2^j}\right) = \sqrt{2^j}[\overline{\boldsymbol{G}}(\overline{\boldsymbol{DF}})^j]_{0,n}, \tag{5.4}$$

respectively. One can show that the terms in Equations 5.3 and 5.4 are given by

$$[(\overline{\boldsymbol{DF}})^j]_{0,n} = [\boldsymbol{f}^\dagger * \boldsymbol{U}\boldsymbol{f}^\dagger * \cdots * \boldsymbol{U}^{j-1}\boldsymbol{f}^\dagger]_n$$

and

$$[\overline{\boldsymbol{G}}(\overline{\boldsymbol{DF}})^j]_{0,n} = [\boldsymbol{f}^\dagger * \boldsymbol{U}\boldsymbol{f}^\dagger * \cdots * \boldsymbol{U}^{j-1}\boldsymbol{f}^\dagger * \boldsymbol{U}^j\boldsymbol{g}^\dagger]_n$$

respectively.

Before computing the scaling function and the approximating wavelet we still need to resolve three issues: (1) Which à trous filter f_k are we going to use?, (2) At what rate are we going to sample $\psi(t)$ to obtain g_k?, and (3) How many samples of $\psi(t)$ are going to be used to define g_k?

Table 5.1: Filter coefficients of Lagrange à trous filters of various lengths P ($2Q-1$ is the degree of the corresponding Lagrange polynomial, $P=4Q-1$).

f	$P=3,\ Q=1$	$P=7,\ Q=2$	$P=11,\ Q=3$	$P=15,\ Q=4$
f_0	$\sqrt{2}/2$	$\sqrt{2}/2$	0.7071	0.7071
$f_{\pm1}$	$\sqrt{2}/4$	$9\sqrt{2}/32$	0.4143	0.4230
$f_{\pm2}$		0	0	0
$f_{\pm3}$		$-\sqrt{2}/32$	-0.0691	-0.0846
$f_{\pm4}$			0	0
$f_{\pm5}$			0.0083	0.0169
$f_{\pm6}$				0
$f_{\pm7}$				-0.0017

The only restriction in choosing the à trous filter is that it must lead to a reasonable behaved scaling function. We arbitrarily adopt the commonly used Lagrange à trous filters [She92]. Table 5.1 shows the filter coefficients of Lagrange à trous filters of various lengths. With regards to the sampling of $\psi(t)$ we will use Nyquist criterion. Since the sine-Gabor wavelet is neither bandlimited nor time limited we need to develop criteria for determining the sampling frequency as well as the number of samples to be used. We use the mean and standard deviation of the wavelet and its Fourier transform to establish such criteria. In the case of sine-Gabor wavelet we refer to: (1) $\sigma(\psi)$ (see Equation 4.2) as the "support" of the wavelet, (2) $m(\hat{\psi})$ (see Equation 4.4) as the center frequency of the wavelet, and (3) $\sigma(\hat{\psi})$ (see Equation 4.3) as the "bandwidth" of the wavelet. It is easy to verify that for $\sigma_0\omega_0 \geq 5/2$ these quantities can be approximated by

$$\sigma(\psi) \approx \frac{\sigma_0}{\sqrt{2}},$$

$$m(\hat{\psi}) \approx \omega_0,$$

and

$$\sigma(\hat{\psi}) \approx \frac{1}{\sqrt{2}\sigma_0}.$$

We determine the maximum frequency for the sine-Gabor wavelet by

$$\omega_{max} = m(\hat{\psi}) + d_\omega \sigma(\hat{\psi}),$$

where the value d_ω is an integer such that the energy of $\hat{\psi}(\omega)$ in the interval $[-\omega_{max}, \omega_{max}]$ is larger than a certain percentage of its total energy. Similarly, we truncate the sine-Gabor wavelet to the time interval

$$[-d_t\sigma(\psi), d_t\sigma(\psi)],$$

where the value d_t is an integer such that the energy of $\psi(t)$ in the above interval is larger than a certain percentage of its total energy.

In order to demonstrate this procedure consider computing the approximating wavelet $\psi'(t)$ for the sine-Gabor wavelet $\psi(t)$ with $\sigma_0 = 5/2$ and $\omega_0 = 1$. Recall that these values lead to a wavelet that is within 1% of the optimum time-frequency resolution. First we must choose a sampling rate. Although the Fourier transform of the sine-Gabor wavelet is not bandlimited one may verify that for $d_\omega = 3$ more than 99.85% of the energy of $\hat{\psi}(\omega)$ is confined to the interval $\omega \in [-1.85, 1.85]$. Therefore, a sampling frequency $\omega_s \geq 3.7$ should suffice in practice. This implies that the sampling period $T_s \leq 1.69$. For simplicity we choose $T_s = 1$. Figure 5.1 shows the sine-Gabor wavelet and its sampled version for $T_s = 1$. Next we must truncate the number of samples to be used. One may verify that for $d_t = 3$ more than 99.84% of the energy of $\psi(t)$ is confined to the interval $t \in [-5.37, 5.37]$. Therefore, we define $g_k = \overline{\psi(-k)}$, $-6 \leq k \leq 6$, $k \in \mathbb{Z}$. Table 5.2 shows the filter coefficients $g_k = \overline{\psi(-k)}$, $-6 \leq k \leq 6$, $k \in \mathbb{Z}$ for $\sigma_0 = 5/2$ and $\omega_0 = 1$. As an example we compute the iterated scaling function and approximating wavelet with g_k as defined above using the Lagrange à trous filter of length $P = 7$. Figures 5.2 and 5.3 show the scaling function and the approximating wavelet for six iterations. The error between $\psi'_{(j)}\left(\frac{n}{2^j}\right)$ and $\psi\left(\frac{n}{2^j}\right)$ for the tenth iteration is shown in Figure 5.4. Notice that indeed $\psi'(k) - \psi(k) = 0$ for $-6 \leq k \leq 6$, $k \in \mathbb{Z}$ as expected (see Equation 3.29). Numerical

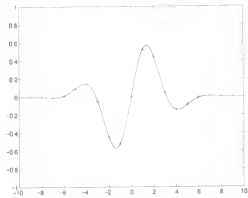
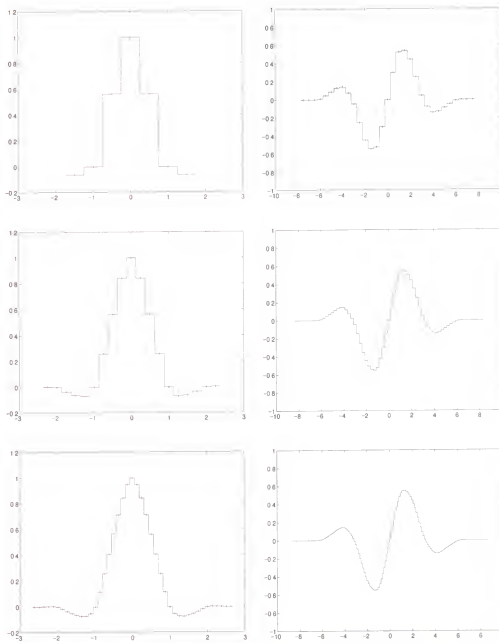


Figure 5.1: Sine-Gabor wavelet for $\sigma_0 = 5/2$ and $\omega_0 = 1$ and corresponding samples for $T_s = 1$.

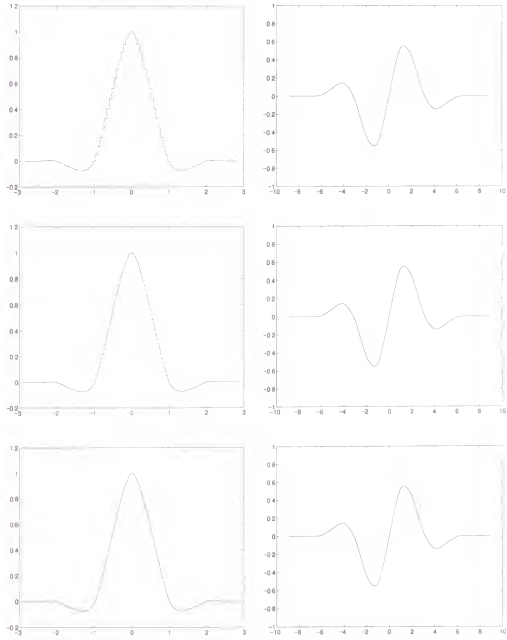
Table 5.2: Filter coefficients $g_k = \overline{\psi(-k)}$ for $\sigma_0 = 5/2$ and $\omega_0 = 1$.

$\boldsymbol{g}$	$M = 13$
g_0	0
$g_1, -g_{-1}$	-0.5224
$g_2, -g_{-2}$	-0.4440
$g_3, -g_{-3}$	-0.0462
$g_4, -g_{-4}$	0.1415
$g_5, -g_{-5}$	0.0873
$g_6, -g_{-6}$	0.0105



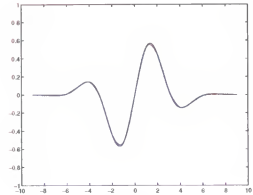
(a) (b)

Figure 5.2: Iterations 1 through 3 for (a) The scaling function $\phi(t)$, and (b) The approximating wavelet $\psi'(t)$. Samples are shown with a '+' sign scale permitting.

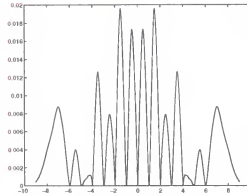


(a) (b)

Figure 5.3: Iterations 4 through 6 for (a) The scaling function $\phi(t)$, and (b) The approximating wavelet $\psi'(t)$.



(a)



(b)

Figure 5.4: Comparison of $\psi'_{(10)}(t)$ and $\psi(t)$. (a) $\psi'_{(10)}\left(\frac{n}{2^{10}}\right)$ (red) and $\psi\left(\frac{n}{2^{10}}\right)$ (blue). (b) $\left|\psi'_{(10)}\left(\frac{n}{2^{10}}\right) - \psi\left(\frac{n}{2^{10}}\right)\right|$. In both cases $-9 \leq \frac{n}{2^{10}} \leq 9$, $n \in \mathbb{Z}$.

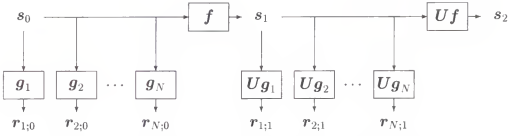


Figure 5.5: Block diagram of a 2-level forward multivoice discrete undecimated wavelet transform.

results show that the relative integrated root mean square error between $\psi'_{(10)}(t)$ and $\psi(t)$, i.e., $\frac{\|\psi'_{(10)}(t)-\psi(t)\|_2^2}{\|\psi(t)\|_2^2}$, is 0.096%.

5.2 Forward Discrete Sine–Gabor Wavelet Transform

In this section we address the implementation issues regarding the forward discrete sine-Gabor wavelet transform. In particular we are interested in computing a multivoice discrete undecimated wavelet transform. An example of a 2-level transform is shown in Figure 5.5. Since our implementation will be carried out on a digital computer we need to constrain input signals s_k to have finite length. Without loss of generality we will assume that s_k is defined for $k \in [0, K-1]$. We take advantage of this limitation to constrain the length of the output signals of the filter bank to have the same length as the input. This is accomplished by using as input the periodic extension of s_k. As a result each of the output signals of the filter bank is periodic with the same period of the input and therefore can be represented with only K samples. However, this limits the length of the original input signal to be at least as long as the longest filter in the filter bank. The reason for this constrain is that we want to preserve the shape of the filter kernels. If the original input signal were to be shorter than a certain filter kernel then the periodic extension of the signal would amount to a wrap around of the filter kernel. In such a case the output of the filter bank will bear no relation to the wavelet "series" coefficients we are seeking

to compute due to the change in the filter kernels. This restriction can be viewed as a limit to the number of levels that can be computed in practice. Let M be the maximum filter length amongst the filters $\boldsymbol{f}, \boldsymbol{g}_1, \ldots, \boldsymbol{g}_N$, and let K be the length of the original input signal, then the maximum number of levels L that can be computed is given by

$$L = \left\lfloor \log_2 \frac{K-1}{M-1} \right\rfloor + 1. \tag{5.5}$$

It can be shown that given a signal s_k, $k \in [0, K-1]$ the above implementation would amount to limiting the number of levels to L or less, using s_k as input, and performing circular convolution instead of convolution in the filter bank. One drawback of this method is that it may introduce artifacts in the outputs of the filter bank as extending the input signal periodically can result in an abrupt transition not present in the original signal. This problem can be alleviated by using as input signal the mirror extension of the original input signal. Given s_k, $k \in [0, K-1]$ its mirror extension m_k is given by

$$m_k = \begin{cases} s_{-(k+1)}, & k = -K, \ldots, -1. \\ s_k, & k = 0, \ldots, K-1. \end{cases}$$

Although from now on we will assume that the input to the filter bank is the mirror extension of the original input s_k we still limit the number of levels to be computed to L in Equation 5.5. Before giving a justification for maintaining this constrain we need to study the signals at the outputs of the filter bank.

Following a scheme similar to the one presented in [Kor96, Opp89], we define the following generalized symmetric/antisymmetric signal types:

- A signal s_k, $k \in [s_s, s_s + K - 1]$ satisfying $s_{s_s+k} = s_{s_s+K-1-k}$, $k \in [0, \lfloor \frac{K-1}{2} \rfloor]$ is said to be of Type I if K is odd and of Type II if K is even.

- A signal s_k, $k \in [s_s, s_s + K - 1]$ satisfying $s_{s_s+k} = -s_{s_s+K-1-k}$, $k \in [0, \lfloor \frac{K-1}{2} \rfloor]$ is said to be of Type III if K is odd and of Type IV if K is even.

Signals of Type I and II are symmetric whereas signals of Type III and IV are antisymmetric. The center of symmetry/antisymmetry is given by $c_s = \frac{K-1}{2} + s_s$. Notice that for Type II and Type IV signals the center of symmetry/antisymmetry is in between two samples.

Given the above signal types the following observations are in order:

1. A mirror extended signal is a signal of Type II.
2. The impulse response of filter $\boldsymbol{U}^j \boldsymbol{f}$ is a signal of Type I.
3. The impulse responses of filters $\boldsymbol{U}^j \boldsymbol{g}_v$ are signals of Type III.
4. The circular convolution of a Type II signal with a Type I filter yields a Type II signal.
5. The circular convolution of a Type II signal with a Type III filter yields a Type IV signal.

As a consequence of these results we have that the $\boldsymbol{s}_j$ are always of Type II and the $\boldsymbol{r}_{v,j}$ are always of Type IV as long as the input signal is of Type II. The importance of this observation resides in the fact that even though the mirror extended signal is $2K$ samples long, only K samples are needed for computing and storing the outputs of the filter bank.

Now we are in a position to give a justification for limiting the number of levels to L even when we use as input the periodic extension of a mirror signal. The "extended" input signals discussed so far are better viewed as boundary conditions to constrain the output signals of the filter bank to have the same length as the input. In the case of the periodic extension it is clear that if the fundamental period of the input signal is shorter than some filter kernel then this amounts to a wrap around of the filter coefficients. A similar result holds true if the input signal is the periodic extension of a mirror signal. If some filter is longer than half the length

of the fundamental period then this amounts to a reflection of the filter coefficients extending beyond half the signal length. Since Equation 5.5 is based on the length of the original signal it is valid in both cases.

Now we turn into a specific example and show how this method can be used. For the purposes of this example we will focus on a sine-Gabor wavelet with $\sigma_0 = 4$ and $\omega_0 = \frac{3\pi}{4}$. This leads to a narrow-band wavelet with $\sigma(\hat{\psi}) = 0.1768$ and relative bandwidth of 0.0750, $\mathcal{Q} = 13.3333$[1]. In order to generate a nearly tight frame ($B/A = 1.0001$) we need in this case at least 8 voices ($N = 8$). Next, we need to sample and truncate each of the wavelets $\psi_1(t), \ldots, \psi_8(t)$. To achieve this we extend the procedure described in Section 5.1. In the case of multiple voices, the mean and standard deviation of the sine-Gabor wavelet and its Fourier transform are given by

$$\sigma(\psi_v) = 2^{(v-1)/N}\sigma(\psi),$$

$$m(\hat{\psi}_v) = 2^{-(v-1)/N}m(\hat{\psi}),$$

and

$$\sigma(\hat{\psi}_v) = 2^{-(v-1)/N}\sigma(\hat{\psi}),$$

where $\psi_v(t)$ is defined by Equation 2.35. We determine the maximum frequency for the multivoice sine-Gabor wavelet by

$$\omega_{max} = m(\hat{\psi}_1) + d_\omega\sigma(\hat{\psi}_1),$$

where the value d_ω is such that the energy of $\hat{\psi}_1(\omega)$ in the interval $[-\omega_{max}, \omega_{max}]$ is larger than a certain percentage of its total energy. Similarly, we truncate the sine-Gabor wavelet $\psi_v(t)$ to the time interval

$$[-d_t\sigma(\psi_v), d_t\sigma(\psi_v)],$$

[1] $\mathcal{Q}$ is the inverse of the relative bandwidth. A filter is said to be narrow-band if $\mathcal{Q} \gg 1$; broad-band if $\mathcal{Q} \approx 1$.

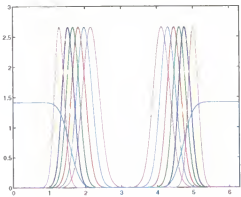


Figure 5.6: Magnitude of the discrete Fourier transform of the filters $\boldsymbol{g}_v$ in Table 5.3 and the Lagrange à trous filter $\boldsymbol{f}$ in Table 5.4.

where the value d_t is such that the energy of $\psi_v(t)$ in the above interval is larger than a certain percentage of its total energy.

One may verify that for $d_\omega = 4$ more than 99.99% of the energy of $\hat{\psi}_1(\omega)$ is confined to the interval $\omega \in [-3.07, 3.07]$. Therefore a sampling period $T_s = 1$ would be sufficient. Notice that the bandwidth of wavelets $\psi_2(t), \ldots, \psi_8(t)$ is smaller than the bandwidth of $\psi_1(t)$ and therefore the same sampling period would be also adequate. Next, we need to determine the number of samples that will be used to define each $g_{v;k} = \psi_v(-k)$. (Notice that we have dropped the complex conjugate from the definition of $g_{v;k}$ since the wavelets in our study are real.) As before we use an energy criterion. One may verify that for $d_t = 4$ more than 99.99% of the energy of $\psi_v(t)$ is confined to the interval $t \in [-2^{(v-1)/N}11.32, 2^{(v-1)/N}11.32]$. Table 5.3 shows the filter coefficients for $g_{v;k}$ for $k \in [-M_v, M_v]$ where $M_v = \lceil 2^{(v-1)/N}11.32 \rceil$. Figure 5.6 shows the magnitude of the discrete Fourier transform of the filters $\boldsymbol{g}_v$ in Table 5.3 and the Lagrange à trous filter $\boldsymbol{f}$ in Table 5.4. Figure 5.7 shows the impulse response of the filter bank in Figure 5.5 for the filter coefficients in Tables 5.3 and 5.4.

Table 5.3: Filter coefficients $g_{v;k} = \psi_v(-k)$ for $v = 1, \ldots, 8$, $\sigma_0 = 4$ and $\omega_0 = \frac{3\pi}{4}$.

$\boldsymbol{g}_v$	$v=1$	$v=2$	$v=3$	$v=4$	$v=5$	$v=6$	$v=7$	$v=8$
$g_{v;0}$	0	0	0	0	0	0	0	0
$g_{v;1}$, $-g_{v;-1}$	-0.3640	-0.3943	-0.4006	-0.3899	-0.3681	-0.3396	-0.3078	-0.2753
$g_{v;2}$, $-g_{v;-2}$	0.4687	0.4053	0.2992	0.1797	0.0668	-0.0281	-0.1006	-0.1511
$g_{v;3}$, $-g_{v;-3}$	-0.2835	-0.0759	0.1218	0.2563	0.3131	0.3034	0.2496	0.1741
$g_{v;4}$, $-g_{v;-4}$	0.0000	-0.2255	-0.3128	-0.2534	-0.1088	0.0478	0.1662	0.2272
$g_{v;5}$, $-g_{v;-5}$	0.1719	0.2478	0.1191	-0.0859	-0.2258	-0.2423	-0.1583	-0.0321
$g_{v;6}$, $-g_{v;-6}$	-0.1724	-0.0732	0.1265	0.2089	0.1158	-0.0547	-0.1807	-0.2051
$g_{v;7}$, $-g_{v;-7}$	0.0812	-0.0740	-0.1459	-0.0249	0.1372	0.1728	0.0686	-0.0769
$g_{v;8}$, $-g_{v;-8}$	0.0000	0.0906	0.0154	-0.1150	-0.0954	0.0501	0.1522	0.1204
$g_{v;9}$, $-g_{v;-9}$	-0.0299	-0.0325	0.0635	0.0546	-0.0693	-0.1100	-0.0055	0.1151
$g_{v;10}$, $-g_{v;-10}$	0.0233	-0.0132	-0.0402	0.0402	0.0642	-0.0386	-0.1038	-0.0317
$g_{v;11}$, $-g_{v;-11}$	-0.0086	0.0198	-0.0060	-0.0392	0.0283	0.0625	-0.0243	-0.0941
$g_{v;12}$, $-g_{v;-12}$	0.0000	-0.0079	0.0181	-0.0053	-0.0360	0.0256	0.0575	-0.0218
$g_{v;13}$, $-g_{v;-13}$		-0.0011	-0.0062	0.0177	-0.0087	-0.0317	0.0290	0.0505
$g_{v;14}$, $-g_{v;-14}$			-0.0030	-0.0032	0.0171	-0.0148	-0.0251	0.0356
$g_{v;15}$, $-g_{v;-15}$				-0.0053	0.0016	0.0143	-0.0218	-0.0146
$g_{v;16}$, $-g_{v;-16}$					-0.0069	0.0076	0.0077	-0.0266
$g_{v;17}$, $-g_{v;-17}$					0.0002	-0.0058	0.0125	-0.0030
$g_{v;18}$, $-g_{v;-18}$						-0.0034	-0.0008	0.0129
$g_{v;19}$, $-g_{v;-19}$							-0.0058	0.0067
$g_{v;20}$, $-g_{v;-20}$							-0.0010	-0.0038
$g_{v;21}$, $-g_{v;-21}$								-0.0046

Table 5.4: Filter coefficients of the Lagrange à trous filter $\boldsymbol{f}$ of length P=39.

$\boldsymbol{f}$	$P = 39$
$f_{\pm 0}$	7.0711e-01
$f_{\pm 1}$	4.3905e-01
$f_{\pm 2}$	0
$f_{\pm 3}$	-1.1974e-01
$f_{\pm 4}$	0
$f_{\pm 5}$	4.7896e-02
$f_{\pm 6}$	0
$f_{\pm 7}$	-1.8422e-02
$f_{\pm 8}$	0
$f_{\pm 9}$	6.1405e-03
$f_{\pm 10}$	0
$f_{\pm 11}$	-1.6747e-03
$f_{\pm 12}$	0
$f_{\pm 13}$	3.5426e-04
$f_{\pm 14}$	0
$f_{\pm 15}$	-5.4181e-05
$f_{\pm 16}$	0
$f_{\pm 17}$	5.3119e-06
$f_{\pm 18}$	0
$f_{\pm 19}$	-2.5014e-07

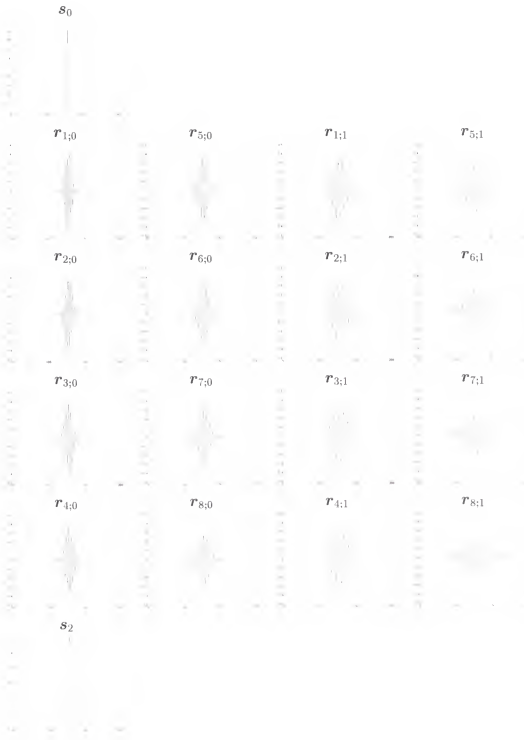


Figure 5.7: Response of the filter bank in Figure 5.5 for the impulse signal in the top left corner and the filter coefficients in Tables 5.3 and 5.4.

5.3 Inverse Discrete Sine–Gabor Wavelet Transform

In this section we address technical and implementation issues regarding the inverse discrete sine-Gabor wavelet transform. A number of conditions for the forward transform to be invertible were reviewed in Chapter 3. We start by making the observation that for the forward transform to be invertible the discrete Fourier transform (DFT) of the filters $\boldsymbol{f}$, $\boldsymbol{g}_v$ cannot be all zero for any given ω; for otherwise the entire frequency content of the signal is not covered by the transform. It is clear that this condition is not satisfied by the filters in Figure 5.6. Therefore, the forward transform using these filters is not invertible. This might be a source of confusion since we showed that the sine-Gabor function is a wavelet frame, i.e., it is complete and stable. The problem evidenced in Figure 5.6 arises from the discretization of the continuous wavelet transform described in Chapter 3. As we already pointed out in Section 5.2, sampling $\psi_v(t)$ leads to filters $\boldsymbol{g}_v$ which are of Type III. It is easy to verify that the DFT of such filters is always zero at $\omega = \pi$. Furthermore, one may verify that the DFT of the Lagrange à trous filters $\boldsymbol{f}$ is also zero at $\omega = \pi$. One may wish to use an alternative filter $\boldsymbol{f}$ with nonzero response at $\omega = \pi$. However, in order for the composition of DWT^{-1} and DWT to converge rapidly, the response of the alternative filter $\boldsymbol{f}$ at $\omega = \pi$ must be significantly different from zero. This defeats the spirit of the multiresolution approach where the filters $\boldsymbol{f}$, $\boldsymbol{g}_v$ split the frequency content of the signal into low and high frequencies, respectively. In fact, the author is not aware of any such filter $\boldsymbol{f}$ with nonzero response at $\omega = \pi$ leading to a reasonably behaved scaling function. We summarize the above discussion by making the following proposition:

Proposition 5.3.1 *The forward discrete wavelet transform is not invertible if the filter* $\boldsymbol{f}$ *satisfies* $\check{f}(e^{j\pi}) = 0$ *and the filters* $\boldsymbol{g}_v$ *are defined by* $g_{v;k} = \psi_v(-k)$, $k \in [-M_v, M_v]$, *where* $\psi_v(t)$ *is any real odd function, i.e.,* $\psi_v(t) = -\psi_v(-t)$, *generating a wavelet frame.*

Proof: The proof follows from the above discussion. ■

Therefore, it follows from Proposition 5.3.1 that it is not possible to invert the forward discrete sine-Gabor wavelet transform by directly using the techniques reviewed in Chapter 3. In order to overcome this difficulty we modify the definition of the filters $\boldsymbol{g}_v$ as follows

$$g_{v;k} = \psi_v\left(-(k+0.5)T_s\right),\ k \in [-M_v, M_v - 1], \tag{5.6}$$

where $M_v = \left\lceil \frac{2^{(v-1)/N} d_t \sigma(\psi)}{T_s} \right\rceil$. This leads to filters of Type IV whose response is generally nonzero at $\omega = \pi$. Notice that we have included the sampling period as we will be considering samplings other than $T_s = 1$. It is worth pointing out that the alternative sampling in Equation 5.6 is equivalent to the sampling shift strategy used in [Mal92b].

In order to understand the effects of the sampling shift strategy, one may view Equation 5.6 as sampling a shifted version of $\psi_v(t)$, i.e.,

$$g_{v;k} = \psi_v^s\left(-kT_s\right),\ k \in [-M_v, M_v - 1],$$

where $\psi_v^s(t) = \psi_v(t - 0.5T_s)$. Therefore, it follows that the results described in Chapter 3 hold for $\psi_v^s(t)$. Furthermore, it is not difficult to see that the good time-frequency localization properties of $\psi_v(t)$ are preserved in $\psi_v^s(t)$. However, two disadvantages arise from the alternative sampling in Equation 5.6: (1) The elegant decomposition algorithm described in Section 5.2 can no longer be used since the filters $\boldsymbol{g}_v$ are now of Type IV, and (2) The filters $\boldsymbol{g}_v$ are no longer constant phase. Before considering the issues regarding the inverse transform we describe how to successfully evade these two problems.

The first problem can be solved by making the following two observations:

1. The circular convolution of a mirror extended signal of length $2K$ (i.e., a Type II signal) with a Type IV filter of shorter length yields a response r_k satisfying [Kor96]

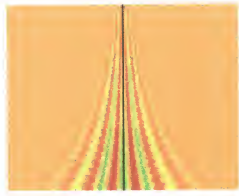
$$r_{s_r+k} = \begin{cases} -r_{s_r+2K-2-k}, & k \in [0, K-1], \\ 0, & k = 2K-1, \end{cases}$$

where s_r is some integer shift. It follows from the above expression that r_{s_r+K-1} is also zero. We will refer to the above response as a modified Type III signal. Notice that although the mirror extended signal is $2K$ samples long, only K and $K-1$ samples are needed to compute and store the output, respectively.

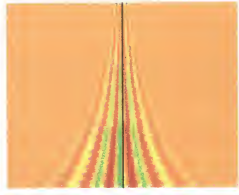
2. For a filter $\boldsymbol{g}_v$ of Type IV one may show that $\boldsymbol{U}^j \boldsymbol{g}_v$ is of Type III for $j > 0$.

From Section 5.2 we know that the $\boldsymbol{s}_j$ are always of Type II. It follows from the two observations above that the $\boldsymbol{r}_{v;0}$ are modified Type III signals and that the $\boldsymbol{r}_{vj}$, $j > 0$ are Type IV signals. Therefore, given a mirror extended signal of length $2K$ is still possible to have an elegant algorithm that requires only K samples for computing the output and either $K-1$ or K samples for storing it.

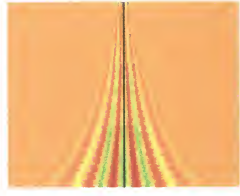
In order to understand the impact of the second problem, Figure 5.8 shows the step response of a 3-level decomposition for the filter coefficients in Table 5.3 and equivalent filter coefficients derived using Equation 5.6 with $\sigma_0 = 4$, $\omega_0 = \frac{3\pi}{4}$, $d_\omega = 4$, $d_t = 4$, and $T_s = 1$. It is clear from the figure that by loosing the constant phase, the filters defined by Equation 5.6 introduce a shift in time that makes tracing the evolution of the coefficients in the time-frequency plane more difficult. However, this problem can be easily evaded by noting that the phase of filter $\boldsymbol{U}^j \boldsymbol{g}_v$ is given by $2^{j-1}\omega - \pi/2$. For $j > 0$ the linear part corresponds to an integer shift (to the left) of 2^{j-1}. This can be compensated for by shifting the coefficients to the right by the same amount (see Figure 5.8(c)). For $j = 0$ the shift is not an integer and



(a)



(b)



(c)

Figure 5.8: Step response of a 3-level/8-voice filter bank. The horizontal axis represents time. The vertical axis represents the filter output. From top to bottom the $v + jN$ row corresponds to the filter output $\boldsymbol{r}_{v;j}$, where N is the number of voices. The black vertical line across the middle of the image denotes the location of the step. (a) Using filters from Table 5.3. (b) Using equivalent filters defined by Equation 5.6. (c) Using the same filters as in (b) with shift compensation for analysis purposes (see text).

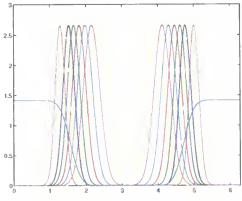


Figure 5.9: Magnitude of the discrete Fourier transform of the equivalent filters $\boldsymbol{g}_v$ defined by Equation 5.6 and the Lagrange à trous filter $\boldsymbol{f}$ in Table 5.4.

we opt to leave the coefficients unchanged. Figure 5.9 shows the magnitude of the discrete Fourier transform of the equivalent filters $\boldsymbol{g}_v$ defined by Equation 5.6 and the Lagrange à trous filter $\boldsymbol{f}$ in Table 5.4.

The issue of inverting the transform can now be addressed since the entire frequency line is covered by the filters $\boldsymbol{f}, \boldsymbol{g}_v$ (see Figure 5.9). Based on Equation 3.52, we consider the inverse double integral type filters

$$\tilde{\boldsymbol{f}} = \boldsymbol{\delta}/\sqrt{2} \text{ and } \tilde{\boldsymbol{g}}_v = c\boldsymbol{g}_v^{\dagger}, \tag{5.7}$$

where the value of c is determined by finding the minimum of the absolute value of the difference of Equation 3.54 and one using the Simplex search method. Notice that the factor $\frac{1}{2^{(v-1)/N}}$ in Equation 3.52 is not included here because the filters $\boldsymbol{g}_v$ are defined in terms of $\psi_v(t)$ in Equation 2.35 which includes an extra factor $\frac{1}{\sqrt{2^{(v-1)/N}}}$ when compared to the definition of $\psi_v(t)$ used in Section 3.4. As discussed in Section 3.4 the filters in Equation 5.7 are only an approximate inverse for the discrete wavelet transform. In order to evaluate the quality of the inverse filters we measure the error between the input, $\boldsymbol{s}$, and output, $\tilde{\boldsymbol{s}}$, of the composition of DWT^{-1} and DWT for a given input signal. For this purpose we use the normalized error defined by [She93]

$$e_n = \left| \frac{s_n}{\|\boldsymbol{s}\|} - \frac{\tilde{s}_n}{\|\tilde{\boldsymbol{s}}\|} \right|,$$

and evaluate the root mean square error given by [She93]

$$e_{rms} = \left(\sum_n e_n^2 \right)^{1/2} = \frac{\|\boldsymbol{s} - \tilde{\boldsymbol{s}}'\|}{\|\boldsymbol{s}\|},$$

where $\tilde{\boldsymbol{s}}' = (\|\boldsymbol{s}\|/\|\tilde{\boldsymbol{s}}\|)\tilde{\boldsymbol{s}}$. In our analysis we use an impulse as input signal and limit the number of levels of decomposition/reconstruction to one.

Results show that for the filters discussed above the mean square error is approximately 0.4960. The speed of convergence of Neumann iterations (see Section 3.4.3) is also relatively slow; the error after 100 iterations with $\mu = 0.5$ is approximately 0.2947. The main reason for this poor performance is that the DFT of the composition of DWT^{-1} and DWT for the filters under consideration is nearly zero around $\omega = \pi$. The only way to improve the performance is to sample at a lower rate; therefore increasing the amount of aliasing and consequently the response of the composition at $\omega = \pi$.

Sampling at a lower rate amounts to selecting a smaller value for d_ω. Filters given by Equations 5.6 and 5.7 with $\sigma_0 = 4$, $\omega_0 = \frac{3\pi}{4}$, $d_\omega = 1.5$, $d_t = 4$, and $T_s = 1.1985$ yield a reduced mean square error of approximately 0.1425. The speed of convergence in this case is also significantly faster; the error after 5 iterations with $\mu = 0.5$ is approximately 0.0223. The magnitude of the discrete Fourier transform of the above filters is shown in Figure 5.10. The output of the composition and the response after 5 iterations using an impulse as input are shown in Figure 5.11.

Clearly, sampling the wavelets $\psi_v(t)$ at a lower rate introduces a certain amount of aliasing in the higher frequency filters $\boldsymbol{g}_1$ and $\boldsymbol{g}_2$ (see Figure 5.10). Consequently, the approximating wavelets $\psi_1'(t)$ and $\psi_2'(t)$ generated by these filters are no longer "close" to the original wavelets $\psi_1(t)$ and $\psi_2(t)$. Figure 5.12 shows the discrete filters $\boldsymbol{g}_v$, the approximating wavelets $\psi_v'(t)$, and the original wavelets $\psi_v(t)$.

From the above analysis we observe that there is a trade-off between the quality of the inverse filters and the speed of convergence, and the quality of the higher

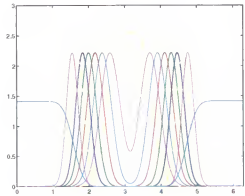
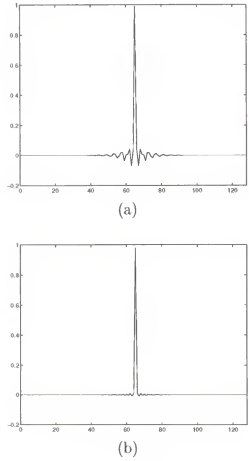


Figure 5.10: Magnitude of the discrete Fourier transform of the equivalent filters $\boldsymbol{g}_v$ defined by Equation 5.6 with $\sigma_0 = 4$, $\omega_0 = \frac{3\pi}{4}$, $d_\omega = 1.5$, $d_t = 4$, and $T_s = 1.1985$, and the Lagrange à trous filter $\boldsymbol{f}$ in Table 5.4.



(a)

(b)

Figure 5.11: Output of the composition of DWT^{-1} and DWT for the filter in Figure 5.10. (a) No iterations. (b) After 5 iterations.

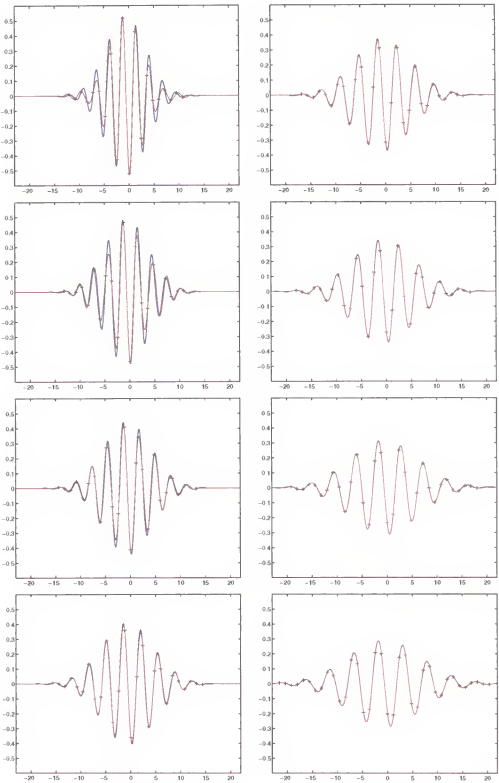


Figure 5.12: Discrete filters $\boldsymbol{g}_v$ ('+'), 4 iterations of the approximating wavelets $\psi_v'(-(t+0.5T_s))$ (red) and original wavelet $\psi_v(-(t+0.5T_s))$ (blue) with $\sigma_0 = 4$, $\omega_0 = \frac{3\pi}{4}$, $d_\omega = 1.5$, $d_t = 4$, and $T_s = 1.1985$. From top to bottom, left to right, $v = 1$ through $v = 8$.

frequency approximating wavelets. By improving the quality of the inverse filter, we indirectly change the shape of the higher frequency approximating wavelets and consequently their time-frequency localization properties. Clearly, the interpretation of the wavelet coefficients changes for the affected voices since the higher frequency analyzing wavelets are no longer "close" to the original wavelets. In fact, the decomposition can now be viewed as a hybrid wavelet transform in which the analyzing wavelet for each voice is not necessarily a fractionally dilated version of a single mother wavelet. We remark that although the above discussed trade-off and interpretation are original to this work they also apply to the work of Shensa [She93, She96].

CHAPTER 6
APPLICATIONS

In this chapter we investigate the application of the sine-Gabor wavelet transform to image processing. We describe an extension of the transform to two dimensions and demonstrate the use of the transform with both broad-band and narrow-band sine-Gabor wavelets having nearly optimum time-frequency localization. We also revisit the use of the sine-Gabor wavelet as an approximation to the first derivative of a Gaussian and demonstrate its use as a multivoice multiresolution edge detector.

6.1 Extension to Two Dimensions

In order to extend the transform to two dimensions we adopt the approach used by Mallat [Mal89a] and construct a separable two dimensional scaling function $\Phi(x, y) = \phi(x)\phi(y)$ and wavelets

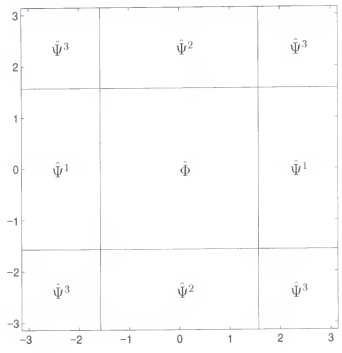
$$\left\{ \Psi^1_{v_x}(x, y) = \psi_{v_x}(x)\phi(y),\ v_x = 1, \ldots, N \right\}, \tag{6.1}$$

$$\left\{ \Psi^2_{v_y}(x, y) = \phi(x)\psi_{v_y}(y),\ v_y = 1, \ldots, N \right\}, \tag{6.2}$$

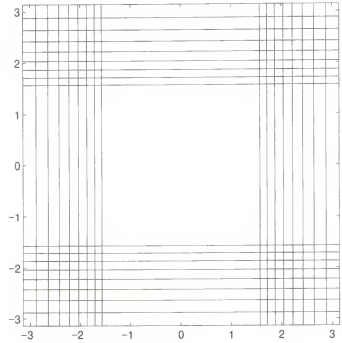
and

$$\left\{ \Psi^3_{v_x, v_y}(x, y) = \psi_{v_x}(x)\psi_{v_y}(y),\ v_x = 1, \ldots, N,\ v_y = 1, \ldots, N \right\},$$

where N is the number of voices. For $N = 1$ one obtains the separable extension to two dimensions of Mallat [Mal89a]; that is, $\Phi(x, y) = \phi(x)\phi(y)$, $\Psi^1(x, y) = \psi(x)\phi(y)$, $\Psi^2(x, y) = \phi(x)\psi(y)$, and $\Psi^3(x, y) = \psi(x)\psi(y)$. Notice that in two dimensions the number of wavelets grows quadratically with respect to the number of voices. It is



(a)



(b)

Figure 6.1: Frequency partition of a separable two dimensional scaling function and wavelets. (a) $N = 1$. (b) $N = 8$.

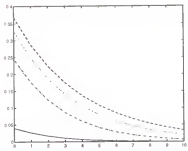


Figure 6.2: Mean square error as a function of the number of iterations for discrete filters $\boldsymbol{g}_v$ with $\sigma_0 = 5/2$, $\omega_0 = 1$, $d_\omega = 1.5$, $d_t = 4$, and $T_s = 2.2080$ and an à trous filter $\boldsymbol{f}$ with $Q = 4$. From top to bottom: $N = 4, 3, 2, 1$.

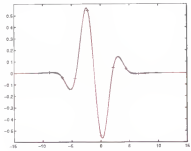


Figure 6.3: Discrete filter $\boldsymbol{g}$ ('+'), 4 iterations of the approximating wavelet $\psi'(-(t + 0.5T_s))$ (red) and original wavelet $\psi(-(t + 0.5T_s))$ (blue) with $\sigma_0 = 5/2$, $\omega_0 = 1$, $d_\omega = 1.5$, $d_t = 4$, and $T_s = 2.2080$.

easy to verify that for N voices the total number of wavelets is $N(N+2)$. Figure 6.1 shows the ideal frequency cover of a separable two dimensional scaling function and wavelets for $N = 1$ and $N = 8$.

In terms of its discrete implementation the two dimensional multivoice wavelet transform using the above separable construction can be achieved by one dimensional filtering operations in the x and y directions. Although tedious due to the excessive number of wavelets for $N > 2$, the discrete implementation of the multivoice wavelet transform is a trivial extension of the classical two dimensional discrete wavelet transform [Mal89a]. To be concise we omit the irksome technical details.

6.2 2–D Broad–Band and Narrow–Band Sine–Gabor Wavelet

In this section we demonstrate the application of both broad-band and narrow-band sine-Gabor wavelets to image processing.

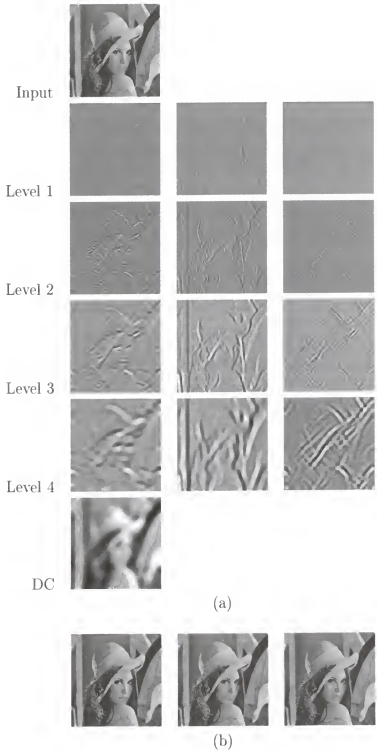


Figure 6.4: Four level decomposition and reconstruction with one voice ($N = 1$) using a discrete filter $\boldsymbol{g}$ with $\sigma_0 = 5/2$, $\omega_0 = 1$, $d_\omega = 1.5$, $d_t = 4$, and $T_s = 2.2080$ and an à trous filter $\boldsymbol{f}$ with $Q = 4$. (a) Decomposition. From left to right: horizontal, vertical, and diagonal channels were applicable. (b) Reconstruction. From left to right: iterations 0, 1, and 2.

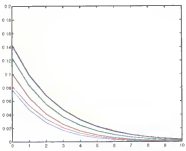


Figure 6.5: Mean square error as a function of the number of iterations for discrete filters $\boldsymbol{g}_v$ with $\sigma_0 = 4$, $\omega_0 = 3\pi/4$, $d_\omega = 1.5$, $d_t = 4$, and $T_s = 1.1985$ and an à trous filter $\boldsymbol{f}$ with $Q = 10$. From top to bottom: $N = 3$ (yellow), $N = 8$ (blue), $N = 7$ (green), $N = 6$ (red), $N = 4$ (magenta), and $N = 5$ (cyan).

As discussed in Chapter 4 the sine-Gabor wavelet with $\sigma_0 = 5/2$ and $\omega_0 = 1$ yields nearly optimum time-frequency localization. It can be verified that $\mathcal{Q} = 3.5355$, i.e., the wavelet is broad-band. Figure 6.2 shows the speed of convergence of the 1–D Neumann iterations as a function of the number of voices for discrete filters $\boldsymbol{g}_v$ with $\sigma_0 = 5/2$, $\omega_0 = 1$, $d_\omega = 1.5$, $d_t = 4$, and $T_s = 2.2080$ and an à trous filter $\boldsymbol{f}$ with $Q = 4$. Notice that the best speed of convergence is obtained using only one voice ($N = 1$). Figure 6.3 shows the approximating wavelet and original wavelet in this case. Even though d_ω is rather small (approximately 93% of the energy of the wavelet is covered in the interval $[-\omega_{max}, \omega_{max}]$) the approximating wavelet is remarkably close to its original counterpart. Figure 6.4 shows a 2–D four level decomposition and reconstruction using the wavelet described above. Observe that the reconstruction is remarkably good even without Neumann iterations.

As discussed in Chapter 5 the sine-Gabor wavelet with $\sigma_0 = 4$ and $\omega_0 = 3\pi/4$ also yields nearly optimum time-frequency localization. It can be verified that $\mathcal{Q} = 13.3333$, i.e., the wavelet is narrow-band. Figure 6.5 shows the speed of convergence of the 1–D Neumann iterations as a function of the number of voices for discrete filters $\boldsymbol{g}_v$ with $\sigma_0 = 4$, $\omega_0 = 3\pi/4$, $d_\omega = 1.5$, $d_t = 4$, and $T_s = 1.1985$ and an à trous filter $\boldsymbol{f}$ with $Q = 10$. Notice that the best speed of convergence is obtained using five voices ($N = 5$). Figure 6.6 shows the approximating and original wavelets in

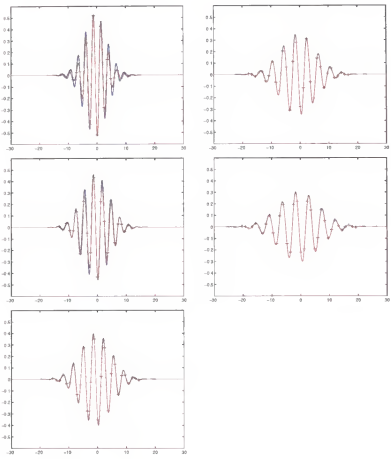


Figure 6.6: Discrete filter $\boldsymbol{g}_v$ ('+'), 4 iterations of the approximating wavelet $\psi'_v(-(t+0.5T_s))$ (red) and original wavelet $\psi_v(-(t+0.5T_s))$ (blue) with $\sigma_0 = 4$, $\omega_0 = 3\pi/4$, $d_\omega = 1.5$, $d_t = 4$, and $T_s = 1.1985$. From top to bottom, left to right, $v = 1$ through $v = 5$.

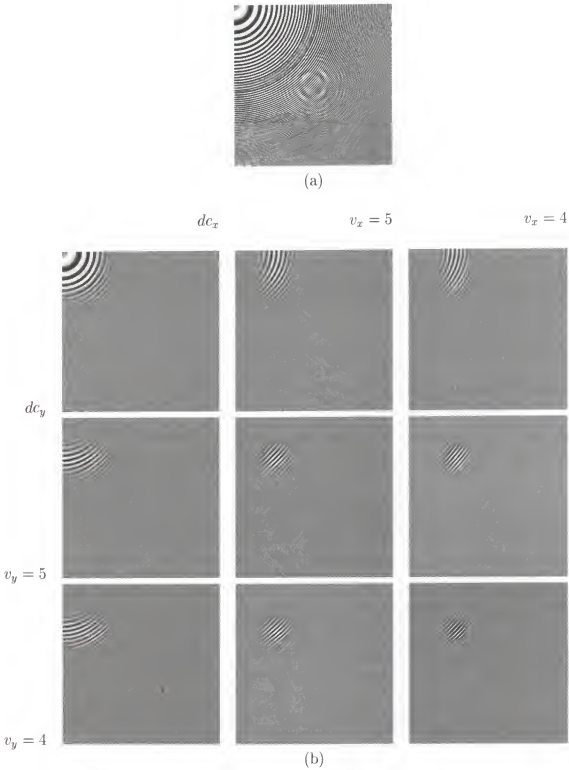
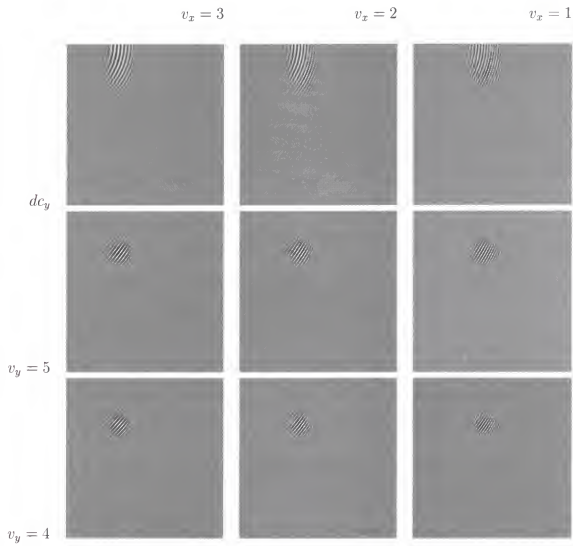
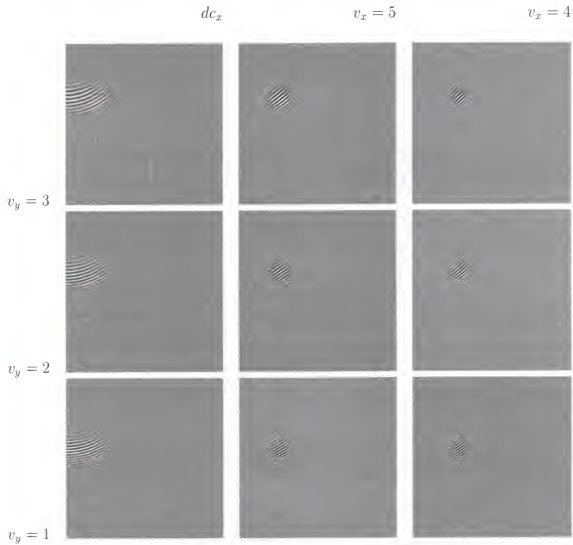


Figure 6.7: Two level decomposition and reconstruction with 5 voices ($N = 5$) using discrete filters $\boldsymbol{g}_v$ with $\sigma_0 = 4$, $\omega_0 = 3\pi/4$, $d_\omega = 1.5$, $d_t = 4$, and $T_s = 1.1985$ and an à trous filter $\boldsymbol{f}$ with $Q = 10$. (a) Input image (2–D chirp). (b) Second level channels. The horizontal and vertical labels (dc_x, $v_x = 5, 4$, and dc_y, $v_y = 5, 4$) indicate the channel shown.



(c)

Figure 6.7: Continued: (c) Second level channels. The horizontal and vertical labels ($v_x = 3, 2, 1$, and dc_y, $v_y = 5, 4$) indicate the channel shown.



(d)

Figure 6.7: Continued: (d) Second level channels. The horizontal and vertical labels (dc_x, $v_x = 5, 4$, and $v_y = 3, 2, 1$) indicate the channel shown.

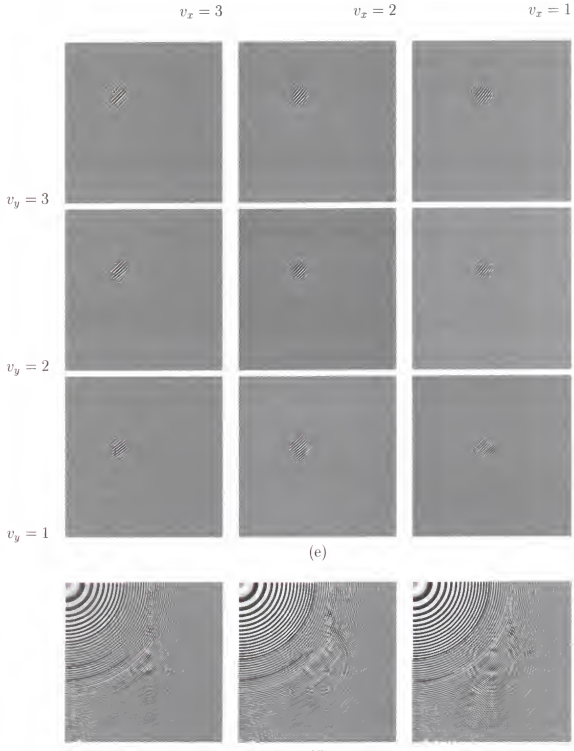


Figure 6.7: Continued: (e) Second level channels. The horizontal and vertical labels ($v_x = 3, 2, 1$, and $v_y = 3, 2, 1$) indicate the channel shown. (f) Reconstruction. From left to right: iterations 0, 2, and 5.

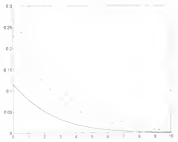


Figure 6.8: Mean square error as a function of the number of iterations for discrete filters $\boldsymbol{g}_v$ with $\sigma_0 = 1.0657$, $\omega_0 = 0.0299$, $d_\omega = 1$, $d_t = 4$, and $T_s = 2.0862$ and an à trous filter $\boldsymbol{f}$ with $Q = 2$. From top to bottom: $N = 3, 2, 1$.

this case. Observe that for $v \geq 3$ the approximating wavelets are remarkably close to their original counterparts. Figure 6.7 shows a 2–D two level decomposition and reconstruction using the wavelet described above.

Based on the above examples and other experimental data we observed that:

1. Broad-band sine-Gabor wavelets are more suitable for applications were good time resolution is required. Usually only one voice is required; although multiple voices can be used to approximate a continuous transform at the expense of slower convergence. Figure 6.4 suggests broad-band sine-Gabor wavelets may be used for boundary detection.

2. Narrow-band sine-Gabor wavelets are more suitable for applications were good frequency resolution is required. Usually multiple voices are required to obtain good convergence. Figure 6.7 suggests narrow-band sine-Gabor wavelets may be used for texture segmentation/classification.

3. By modifying the parameters σ_0 and ω_0 it is possible to change the relative bandwidth (or $\mathcal{Q}$) of the sine-Gabor wavelet transform while preserving nearly optimum time-frequency localization.

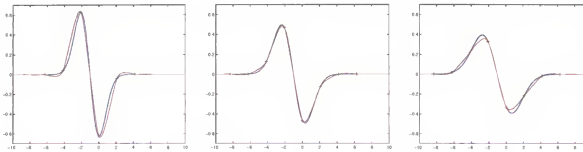


Figure 6.9: Discrete filter $\boldsymbol{g}_v$ ('+'), 4 iterations of the approximating wavelet $\psi'_v(-(t+0.5T_s))$ (red) and original wavelet $\psi'_v(-(t+0.5T_s))$ (blue) with $\sigma_0 = 1.0657$, $\omega_0 = 0.0299$, $d_\omega = 1$, $d_t = 4$, and $T_s = 2.0862$. From top to bottom $v = 1$ through $v = 3$.

6.3 2–D Sine–Gabor Edge Detector

In this section we investigate the use of the sine-Gabor wavelet as an edge detector. Canny [Can86] introduced an optimal detector for 1–D step edges and suggested the first derivative of a Gaussian as an efficient approximation. Image edges of a particular orientation can be detected by convolving the image with a mask generated by convolving a linear edge detection function aligned normal to the edge direction with a projection function parallel to the edge direction[1] [Can86]. Significant computational savings are possible if the projection function is a Gaussian with the same σ_0 as the first derivative of the Gaussian used as the detection function [Can86]. In this case edges of a particular orientation can be detected by convolving the image with a symmetric two dimensional Gaussian and then differentiating normal to the edge direction. In fact this procedure need not be carried out in every direction because the slope of a smooth surface in any direction can be determined exactly from its slope in two directions [Can86]. The operation can be summarized as follows [Can86, Mal92b]:

1. An image $s(x, y)$ is smoothed via convolution with

$$\theta(x, y) = K_G \exp\left(-\frac{x^2 + y^2}{2\sigma_0^2}\right). \tag{6.3}$$

[1]The term edge direction refers to the direction of the tangent to the contour that the edge defines in two dimensions.

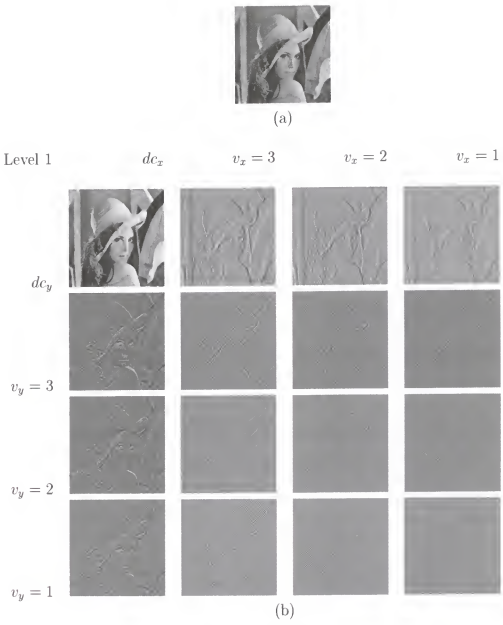
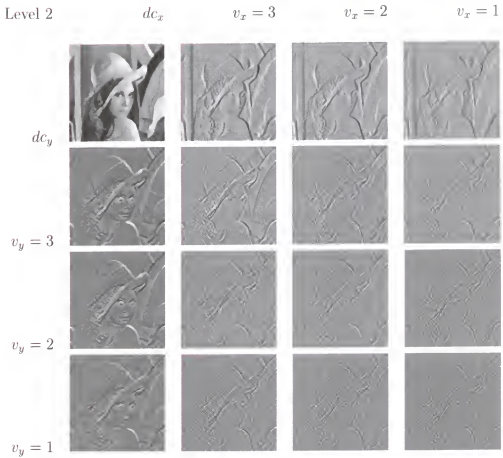


Figure 6.10: Three level decomposition and reconstruction with 3 voices ($N = 3$) using discrete filters $\boldsymbol{g}_v$ with $\sigma_0 = 1.0657$, $\omega_0 = 0.0299$, $d_\omega = 1$, $d_t = 4$, and $T_s = 2.0862$ and an à trous filter $\boldsymbol{f}$ with $Q = 2$. The horizontal and vertical labels (dc_x, $v_x = 3, 2, 1$, and dc_y, $v_y = 3.2, 1$) indicate the channel shown. (a) Input image. (b) Level 1.



(c)

Figure 6.10: Continued: (c) Level 2.

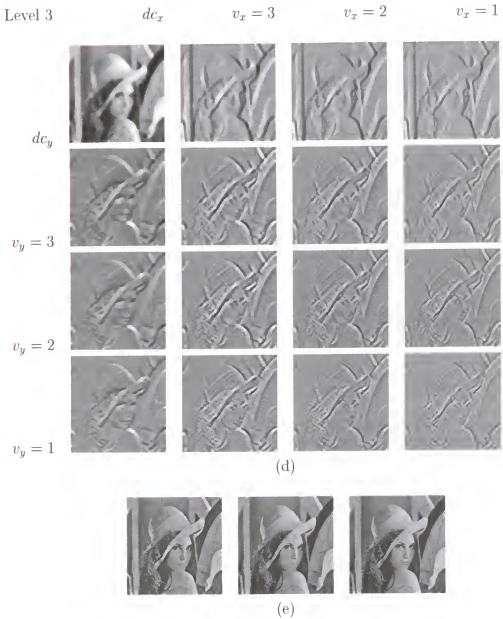


Figure 6.10: Continued: (d) Level 3. (e) Reconstruction. From left to right: iterations 0, 2, and 5.

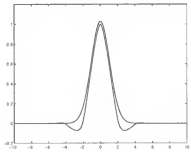
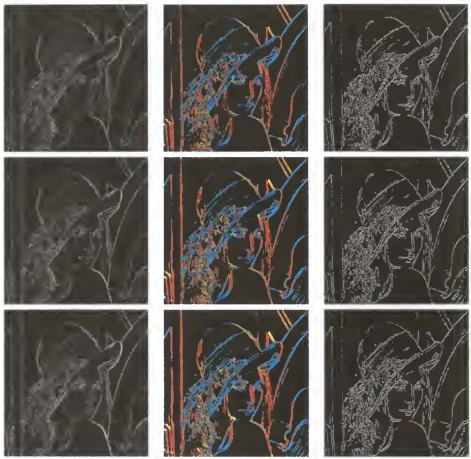
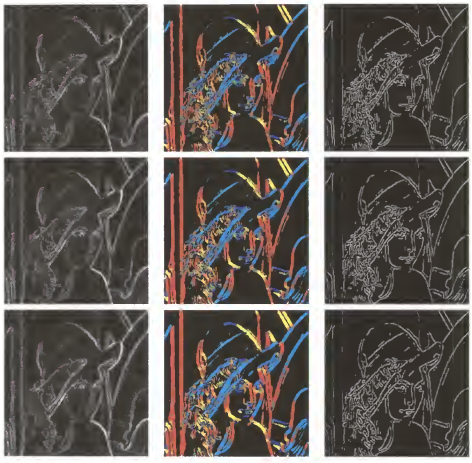


Figure 6.11: Gaussian function generating ψ and the scaling function ϕ for an à trous filter $\boldsymbol{f}$ with $Q = 2$.



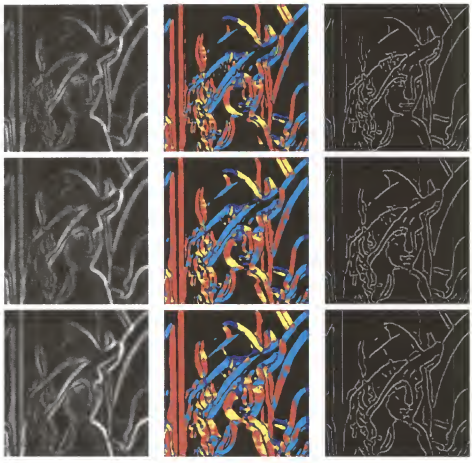
(a)

Figure 6.12: Gradient magnitude, gradient angle, and edges of a three level decomposition with three voices ($N = 3$) using discrete filters $\boldsymbol{g}_v$ with $\sigma_0 = 1.0657$, $\omega_0 = 0.0299$, $d_\omega = 1$, $d_t = 4$, and $T_s = 2.0862$ and an à trous filter $\boldsymbol{f}$ with $Q = 2$. From left to right: magnitude, angle, and edges. From top to bottom: $v = 1, 2, 3$. The angle is shown only when the magnitude is significantly different from zero. Angles within 135 ± 22.5 degrees, 90 ± 22.5 degrees, 45 ± 22.5 degrees, and 0 ± 22.5 degrees are shown in yellow, red, cyan, and blue, respectively. (a) Level 1.



(b)

Figure 6.12: Continued: (b) Level 2.



(c)

Figure 6.12: Continued: (c) Level 3.

2. The gradient vector is then computed

$$\nabla(s * \theta)(x, y) = \left(\left(f * \frac{\partial \theta}{\partial x} \right)(x, y), \left(f * \frac{\partial \theta}{\partial y} \right)(x, y) \right).$$

The direction of the gradient vector at a point (x_0, y_0) indicates the direction in the image plane along which the directional derivative of $f(x, y)$ has the largest absolute value.

3. Edges are defined as points (x_0, y_0) where the modulus of the gradient vector is maximum in the direction of the gradient vector; i.e., edge points are inflection points of the surface $(s * \theta)(x, y)$.

The procedure is usually carried out at different scales to find the best trade-off between detection and localization [Can86].

Mallat [Mal92b] related the above algorithm to the formulation of a 2–D wavelet transform by defining two wavelet functions $\Psi^1(x, y)$ and $\Psi^2(x, y)$ such that

$$\Psi^1(x, y) = \frac{\partial \theta(x, y)}{\partial x} \text{ and } \Psi^2(x, y) = \frac{\partial \theta(x, y)}{\partial y}.$$

The wavelet decomposition using these wavelets is equivalent to a multiscale Canny edge detector [Mal92b]. Due to the separability of $\theta(x, y)$ in Equation 6.3 the wavelets $\Psi^1(x, y)$ and $\Psi^2(x, y)$ may be written as in Equations 6.1 and 6.2 where ϕ and ψ are a Gaussian and its first derivative, respectively.

As discussed in Chapter 4 the sine-Gabor wavelet with $\sigma_0 = 1.0657$ and $\omega_0 = 0.0299$ is for practical purposes equal to the first derivative of a Gaussian (Canny's approximation to the optimal step edge detector [Can86]). Figure 6.8 shows the speed of convergence of the 1–D Neumann iterations as a function of the number of voices for discrete filters $\boldsymbol{g}_v$ with $\sigma_0 = 1.0657$, $\omega_0 = 0.0299$, $d_\omega = 1$, $d_t = 4$, and $T_s = 2.0862$ and an à trous filter $\boldsymbol{f}$ with $Q = 2$. Notice that the best speed of convergence is obtained using only one voice ($N = 1$). However, in order to demonstrate the use of voices we select $N = 3$ at the expense of having slower convergence. Figure 6.9

shows the approximating and original wavelets in this case. Figure 6.10 shows a 2–D three level decomposition and reconstruction using these wavelets. Although ϕ is not a Gaussian we assume the approximation as valid (see Figure 6.11). Figure 6.12 shows the gradient magnitude, gradient angle, and edges using the above filters for the 2–D three level decomposition shown in Figure 6.10. The resulting edge set may be viewed as the output of a multivoice multiresolution Canny edge detector. The addition of voices may prove useful when trancing the evolution of edges through scale-space is required (e.g., singularity detection and processing [Mal92a]).

CHAPTER 7
CONCLUSIONS

In this work we introduced a new nonorthogonal wavelet: the sine-Gabor function. We showed that this function not only satisfies the wavelet admissibility condition but achieves nearly optimum time-frequency localization and generates a frame of wavelets. Furthermore, we showed that there is a trade-off between the tightness of the frame and the time-frequency localization properties of the sine-Gabor wavelet. We demonstrated how this trade-off can be overcome by introducing voices. We also showed that under certain conditions the sine-Gabor wavelet is nearly equal to the first derivative of a Gaussian and generates a "snug" frame. We addressed the implementation issues regarding the computation of the discrete wavelet transform. We showed that there exist a class of wavelets (the sine-Gabor wavelet being one of them) for which the discrete wavelet transform is not invertible. We described how to evade this problem by modifying the strategy for selecting the discrete filters implementing the transform. Furthermore, we showed how in the case of the sine-Gabor wavelet, desirable properties such as constant phase and nearly optimum time-frequency localization are preserved by the alternative filters. We concluded our investigation by extending the transform to two dimensions and demonstrating some of its uses in image processing including a multivoice multiresolution edge detector. However, a number of topics remain to be investigated. Understanding the link between the role of voices in the continuous and discrete implementations would be desirable for filter design purposes. We have seen that although a wavelet may generate a "snug" frame, its discrete implementation is not necessarily invertible or converges rapidly. In particular, the relation between the speed of convergence of

the discrete implementation and the tightness of the wavelet frame needs further study. The author speculates that the role of voices in the discrete case is somewhat different than in the continuous case. Although in the continuous case the addition of voices leads to tighter frames, in the discrete case this is not necessarily the case. We conjecture that using a transform that suppresses a few low frequency voices would yield faster convergence than a transform that uses fewer voices to avoid the overlap between the lowpass filter $\boldsymbol{f}$ and the low frequency voices.

Other areas of research worth pursuing include the development of an efficient algorithm for evaluating Equation 3.31 given a signal $s(t)$ or its sampled version according to Shannon's sampling theorem, and studying the quality of the approximation of the discrete wavelet transform to the exact sampled wavelet transform. Finally, the extension of the proposed methodology for 2–D nonseparable nonorthogonal wavelets would certainly be fruitful for image processing and computer vision applications that use ad-hoc procedures for computing and inverting 2–D nonseparable nonorthogonal wavelet transforms.

REFERENCES

[Bov92] A. C. Bovik, N. Gopal, T. Emmoth, and A. Restrepo (Palacios). Localized measurement of emergent image frequencies by Gabor wavelets. *IEEE Trans. Inf. Theory*, 38(2):691–712, 1992.

[Can86] J. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, 1986.

[Chu92] C. K. Chui. *An Introduction to Wavelets.* Academic Press, Inc., Boston, MA, 1992.

[Com90] J. M. Combes, A. Grossmann, and Ph. Tchamitchian, editors. *Wavelets: Time-frequency Methods and Phase Space*, Berlin; New York, 1990. Springer-Verlag.

[Dau88] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Comm. Pure Appl. Math.*, 41(7):909–996, 1988.

[Dau90] I. Daubechies. The wavelet transform, time-frequency localization and signal analysis. *IEEE Trans. Inf. Theory*, 36(5):961–1005, 1990.

[Dau92] I. Daubechies. *Ten Lectures on Wavelets.* Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992.

[Dau92] I. Daubechies, S. Mallat, and A. S. Willsky, editors. *IEEE Trans. Inf. Theory*, 38(2), New York, 1992. The Institute of Electrical and Electronics Engineers, Inc.

[Gab46] D. Gabor. Theory of communication. *J. Inst. Electr. Eng.*, 93(26):429–457, 1946.

[Gro90] A. Grossmann, R. Kronland-Martinet, and J. Morlet. Reading and understanding continuous wavelet transforms. In J. M. Combes, A. Grossmann, and Ph. Tchamitchian, editors, *Wavelets: Time-frequency Methods and Phase Space*, pages 2–20, Berlin; New York, 1990. Springer-Verlag.

[Hol90] M. Holschneider, R. Kronland-Martinet, J. Morlet, and Ph. Tchamitchian. A real-time algorithm for signal analysis with the help of the wavelet transform. In J. M. Combes, A. Grossmann, and Ph. Tchamitchian, editors, *Wavelets: Time-frequency Methods and Phase Space*, pages 286–304, Berlin; New York, 1990. Springer-Verlag.

[Kor96] I. Koren. *A Multiscale Spline Derivative-Based Transform for Image Fusion and Enhancement.* PhD thesis, University of Florida, December 1996.

[Lee96] T. S. Lee. Image representation using 2D Gabor wavelets. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18(10):959–971, 1996.

[Mal89a] S. G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(7):674–693, 1989.

[Mal89b] S. G. Mallat. Multiresolution approximations and wavelet orthonormal bases of $L^2(\mathbb{R})$. *Trans. Amer. Math. Soc.*, 315(1):69–87, 1989.

[Mal89c] S. G. Mallat. Multifrequency channel decompositions of images and wavelet models. *IEEE Trans. Acoust. Speech Signal Process.*, 37(12):2091–2110, 1989.

[Mal92a] S. Mallat and W. L. Hwang. Singularity detection and processing with wavelets. *IEEE Trans. Inf. Theory*, 38(2), 1992.

[Mal92b] S. Mallat and S. Zhong. Characterization of signals from multiscale edges. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(7), 1992.

[Meh92] R. Mehrotra, K. R. Namuduri, and N. Ranganathan. Gabor filter-based edge detection. *Pattern Recognit.*, 25(12):1479–1494, 1992.

[Opp89] Alan V. Oppenheim and Ronald W. Schafer. *Discrete-Time Signal Processing.* Prentice-Hall, Englewood Cliffs, NJ, 1989.

[Rio91] O. Rioul and M. Vetterli. Wavelets and signal processing. *IEEE Signal Process. Mag.*, 8(4):14–38, 1991.

[She92] M. J. Shensa. The discrete wavelet transform: Wedding the à trous and Mallat algorithms. *IEEE Trans. Signal Process.*, 40(10):2464–2482, 1992.

[She93] M. J. Shensa. An inverse DWT for nonorthogonal wavelets. Technical report, Naval Command, Control and Ocean Surveillance Center, TR 1621, 1993. Available at `ftp.nosc.mil` in `pub/Shensa`.

[She96] M. J. Shensa. Discrete inverses for nonorthogonal wavelet transforms. *IEEE Trans. Signal Process.*, 44(4):798–807, 1996.

[Vet92] M. Vetterli and C. Herley. Wavelets and filter banks: Theory and design. *IEEE Trans. Signal Process.*, 40(9):2207–2232, 1992.

BIOGRAPHICAL SKETCH

Sergio Schuler was born in Caracas, Venezuela, in 1966. He received his BSEE degree from the Universidad Simón Bolívar in 1989, and MSEE and MSCS degrees from the University of Florida in 1991 and 1994, respectively. He will receive his Ph.D. degree from the Department of Computer and Information Science and Engineering at the University of Florida in 1997.

While at the University of Florida he received the President's Spring Recognition of Outstanding Students Award in 1997, the Academic Achievement Award from the College of Engineering in 1996, and the Outstanding Master's Thesis in Electrical Engineering Award in 1992 for his work on the system identification and control of an in-line microwave blood warmer. He also received the Outstanding Undergraduate Thesis in Electrical Engineering Award from Universidad Simón Bolívar in 1989 for his work on the design and implementation of a non-interactive teletext decoder for the NTSC standard.

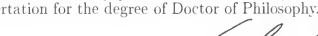
He has been a member of the Tau Beta Pi National Engineering Honor Society since 1993 and the Honor Society of Phi Kappa Phi since 1997.

His research interest include wavelets, signal/image processing, computer vision, computer graphics, software/hardware engineering, computer security and systems engineering.

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Andrew F. Laine, Chairman
Associate Professor of Computer and
Information Science and Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.



Gerhard Ritter
Professor of Computer and
Information Science and Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Richard E. Newman-Wolfe
Assistant Professor of Computer and
Information Science and Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Fred J. Taylor
Professor of Electrical and Computer
Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Murali Rao
Professor of Mathematics

This dissertation was submitted to the Graduate Faculty of the College of Engineering and to the Graduate School and was accepted as partial fulfillment of the requirements for the degree of Doctor of Philosophy.

August 1997

Winfred M. Phillips
Dean, College of Engineering

Karen A. Holbrook
Dean, Graduate School